# An Architecture and Protocol for Communications Satellite Constellations Regarded as Multi-Agent Systems

Craig A. Lindley

CSIRO Division of Information Technology
Locked Bag 17, North Ryde NSW Australia 2113

## Abstract

This paper presents an architecture for satellites regarded as intercommunicating agents. The architecture is based upon a postmodern paradigm of artificial intelligence in which represented knowledge is regarded as text, inference procedures are regarded as social discourse and decision making conventions, and the semantics of representations is grounded in the situated behaviour and activity of agents. A particular protocol is described for agent participation in distributed search and retrieval operations conducted as joint activities.

Keywords: Spacecraft Control, Multi-Agent Systems, Postmodern AI.

## Introduction

Previous work has defined a layered competency model for autonomous spacecraft according to the Subsumption Architecture for intelligent robotics (Lindley, 1993a). Detailed designs for a power control competency (Lindley, 1993b) and a competency concerned with emergent planning and scheduling of spacecraft payload operations (Lindley, 1994) have been developed. These competencies define what may be called "survival" and "service" competencies, respectively. Survival level competencies are concerned with ensuring the survival of a satellite as an autonomous system, while service level competencies are concerned with the provision of operational services to ground-based users of the satellite. When satellite services are provided by satellite constellations, the issue arises as to which satellite or group of satellites should provide services in support of particular user requests. For satellites to autonomously decide how to collectively provide particular services it is necessary for them to be able to inter-negotiate about the availability of resources, the presence of competing demands of differing priorities, and the suitability of different platforms for different tasks. These considerations require a further level of satellite competence that will be called the "social" level. This level allows each satellite to participate in the constellation as a member of a system of agents having collective responsibility for providing users services. The ability to negotiate and participate in joint activities represents a form of "high level" cognitive skill that has traditionally been addressed using world-modelling and deliberative reasoning. However, the work described here begins with a behavioural paradigm, and social level competence is built upon a behavioural foundation. The result is a unified *postmodern* framework for artificial intelligence that integrates a social and hermeneutic conception of knowledge with a behavioural conception of semantics. Within this framework, the essence of intelligence is not representation, but the ability to *use* representations within a social context. The postmodern paradigm extends the advantages of behavioural approaches to high level cognition, unifying the survival, service, and social functions of an autonomous system within a single theoretical model.

In this paper, social-level competencies are explored in relation to a protocol for distributed search and retrieval in a satellite network.

## Postmodern Artificial Intelligence

Representation-centred artificial intelligence has focussed upon the representation of declarative knowledge in the codifications created during knowledge engineering. The "traditional" view in AI research is that declarative knowledge exists prior to knowledge engineering, and that the tools, techniques, and languages of

knowledge engineering are adequate to "extract", "acquire", or "model" that knowledge. Represented knowledge is processed by an algorithm implementing logical inference, and this is taken as a general model of thought. This "logicist" approach to AI (Kirsch, 1991) manifests many of the assumptions of modernist philosophy, and hence can be referred to as "modernist AI".

In the context of autonomous agency the logicist/modernist approach has led to the traditional functional decomposition of the robot control problem resulting in a high level "horizontal" processing sequence that begins with sensation, with data and control flow then passing through perception, modelling, planning, task execution, and finally to actuator control (Brooks, 1986). This form arises from a dependency upon world modelling at the heart of intelligence, since it reflects the sequence of operations required to update and use a world model in an embedded agent.

Logicist AI has been successful in a number of areas, such as automating some forms of expert reasoning, but it has produced brittle, inflexible systems and poor performance when applied to autonomous agents (Maes, 1993). Poor performance in real time arises from the use of inference as a computational process. Brittleness and inflexibility are ubiquitous problems for representation-centred AI, fundamentally associated with their reliance upon models of the world and the difficulty or impossibility of providing sufficient knowledge to deal with real conditions of system operation. Alternative approaches, such as behavioural AI, have sought to avoid these difficulties by ephasising the embedded, situated nature of agents, shifting the emhasis away from representation of the world and towards a conception of agents as dynamic systems that converge toward equilibrium states (goals). The situated nature of intelligent agency also calls for an emphasis upon the social and cultural contexts of knowledge and expertise representing a further departure from modernist epistemology.

Postmodern discourse (Waugh, 1992) provides a meta-discourse for AI research in which many of the limitations of modernist AI are found to be symptomatic of modernism in general. This does not mean that AI research should be abandoned, but suggests a major reappraisal of the expectations placed upon, and role of, AI systems, as well as suggesting alternative models of cognition as a basis for principles of engineering practice. AI researchers can take into account the deep limitations of, and conceptual difficulties with, world representation and reasoning, while nevertheless seeking rigorous principles for the development of useful systems within the instrumentalist terms of reference of engineering science.

Several recent developments in AI research can be integrated into a postmodern approach that differs significantly from a modernist approach. In particular:

- behavioural AI incorporates many postmodern principles concerning the situated nature of cognition, and the deficiencies of reason and representation. It emphasises systems that *act in the world*, providing a pragmatic semantics for systems that engage in "high-level cognition".

- concepts of multi-agent systems and distributed AI incorporate postmodern principles of the cultural situatedness of cognition, and in many cases acknowledge the social nature and function of representation and reasoning. It is in the social sphere that representation and inferential reasoning must be located, as conventions for the intercoordination of behaviour, and hence mechanisms for realising the pragmatic semantics of systems of behavioural agents.

- knowledge engineering must be reconceived as a socially embedded authoring process, aimed at articulating agreed (ie. conventional) codes of practice within particular contexts. The results are available for computational processing, and for the intercoordination of the behaviour of both human and artificial processes.

To accept a postmodern position is to abandon ideas of abstract representation, abstract reasoning, and inferential thought processes as the essential constituents of intelligence. Instead,

*being-in-the-world* becomes the starting point for the analysis of cognition; that analysis must be carried out in full recognition of the situatedness of the analyst, and situatedness must be used as a general concept in the design and structure of computational artefacts. Formal methods of logical analysis and the rationalist methodology of modernism can be adopted as analytical tools, but the broader agenda of the postmodern approach places these in their proper historico-cultural context as tools for *deconstruction* and for the logical synthesis of social codes (processable by computer) for the coordination of behaviour within broader, non-rational, contexts (or forms of life).

The *constructive interaction theory* of Gammack and Anderson (1990) serves as an illustration of what this can mean in the context of knowledge engineering. According to constructive interaction theory, knowledge engineering is an interactive, conversational process, in which the meaning of represented knowledge structures is determined dynamically in the context of knowledge acquisition. The meaning of representations is established with respect to an encompassing reference frame lying outside the data structures, and this reference frame is ultimately never representable (trying to represent it will be infinitely regressive). The outcome of knowledge elicitation can only be viewed "as a specific agreed representation of shared meaning understood by both knowledge engineer and expert". The elicited knowledge is limited to its context of elicitation, and its relevance and applicability to other contexts go unrepresented. Knowledge engineering is a species of conversation, or a *constructive interaction* involving the skills of a domain expert and the knowledge engineer, in which domain knowledge, knowledge of elicitation techniques, and knowledge representation formalisms "interact to produce an agreed representation of domain facts to be coded prior to use by a third party".

It follows that models of knowledge can be used by autonomous agents as resources, much as textual resources are used in human action and problem solving. The coordination of behaviour via texts requires active text interpretation on the part of behavioural agents, during which the influence of text on the form(s) of behaviour constitute the behavioural semantics of that text. The intensional and semantic processes underlying natural intelligence are situated processes of behaviour and action. The construction of autonomous intelligent systems should therefore begin with behaviour, and behavioural possibilities bound the semantics of representations.

The codification of meanings as a product of behaviour, and as a mechanism for behavioural coordination and interaction, is an issue of the language-using competence of a system. Natural language processing and understanding, as planning and behavioural control, have been extensively addressed as symbolic reasoning problems in the modernist AI paradigm. The postmodern AI paradigm provides an alternative approach to language activity in which situated, behavioural agency supplies the semantics of the production and consumption of codifications. This paradigm promises a unified approach to language use in which plans are viewed as communications together with rigorously defined decision models, informal text objects, and other language artefacts.

## Spacecraft and Network Architecture

The AUSTRALIS-1 spacecraft is currently being designed by an informal consortium of Australian universities (ASRI, 1994). The baseline spacecraft is a 35 cm cube, having an expected mass of less than fifty kilograms. It is intended to operate within an altitude range from five hundred to one thousand kilometres. The spacecraft will carry a near infra-red CCD-based camera system, and telecommunications equipment to support a data store-and-forward communications service. AUSTRALIS-1 has particular requirements for autonomy derived from the need to serve a large number of users over a highly dispersed area, using very cheap and simple ground equipment with minimal centralised control or coordination.

AUSTRALIS-1 users will be able to uplink data files, request data broadcast, request image acquisition, and request data deletion. The commands are stored on board the spacecraft, and function as goals within the payload

17

planning and scheduling competency of the autonomous control system. In the data store and forward mode, ground stations can uplink a request for the spacecraft to broadcast data immediately, or broadcast to a distant ground station specified in terms of a latitude and longitude or a time. Stored data is held in an onboard database within the Command Management System (CMS) for downlinking to specified destinations. Stored data can be deleted upon explicit user request. In all of these transactions the spacecraft will schedule and execute operations without coordination or mediation by a central ground station or command and control network. That is, user stations will interact directly with the spacecraft.

In this paper, the basic satellite model used is extended to include intersatellite communication links, presented to the control system as a set of virtual channel interfaces. This represents a major increase in the complexity of the satellite communications system, and has a significant impact upon overall satellite system design, bearing upon the power system capacity, attitude control requirements, thermal and structural design, and on-board computing requirements. For present purposes, these implications will not be elaborated, and the extended capability will be treated as an extension supported by the currently defined platform. While this is inadequate in practice, it is appropriate for considering the principles involved in integrating social-level competencies with more basic survival- and service-level competencies.

The realisation of social level skills requires an interagent communication medium. This can be realised by some combination of ground network connections, ground-to-spacecraft connections, and spacecraft-to-spacecraft connections. GEO satellites within the system can provide indirect satellite-to-satellite connectivity between a large number of LEO satellites, possibly providing continuous and complete intersatellite connectivity for the LEO system, but at the cost of introducing substantial signal propagation delays, increasing the cost and complexity of the LEO satellites, and incurring the cost of the GEO satellites. Hence there are four major types of agents in the proposed multiagent system: ground stations,

ground network nodes that may connect to other network nodes and/or ground stations, LEO satellites, and GEO satellites. Assuming basic Transfer Layer, point-to-point communications services between physical neighbours of the network, the communications system has some stable subnetworks (on the ground and between the ground and GEO satellites), and subnet connections to and between individual LEO satellites that vary continuously, creating a larger scale network having a very dynamic subset of connections. The system is an open one in the sense that additional user nodes or satellites may be attached or removed at any time.

## Multi-Agent Systems

The critical characteristics of agents in multi-agent systems (MAS) that distinguish them from "intelligent" objects and/or processes are the adoption of the terminology and concepts of teleology and social interaction to design the mechanisms of computational process behaviour and interaction, respectively. It is natural to refer to an autonomous satellite as an "agent", since it has goals attributed to it, and the appropriate autonomous coordination and execution of satellite behaviours is required to satisfy those goals. When a number of satellites are available, and user goals could be satisfied by one or more of the satellites, alone or in cooperation, goals become collective and it is necessary to coordinate the behaviours of a number of elements of the system in order to satisfy them. The language and concepts of social interaction naturally arise in the analysis and design of agent cooperation mechanisms for coordinating behaviour to satisfy collective or system-level goals.

Gasser (1991) describes six basic problems that DAI/MAS systems have begun to address that are inherent to the design and implementation of any system of coordinated problem solvers:

1. How to formulate, describe, decompose, and allocate problems, and how to synthesise results among a group of intelligent agents. Suggested bases for decomposition have included abstraction levels, functional, data, or control dependencies, and interaction

density. Participation of an agent in a social activity is typically described as a *commitment* to a *joint activity*.

2. How to enable agents to communicate and interact: what communication languages or protocols to use, and what and when to communicate. Major approaches include formalised interaction and negotiation protocols.

3. How to ensure that agents act coherently in making decisions or taking actions, accommodating the non-local effects of local decisions and avoiding harmful interactions. Major approaches include establishing organisation, improving local awareness and skill, multi-agent planning, abstraction, and resource-directed coherence.

4. How to enable individual agents to represent and reason about the actions, plans, and knowledge of other agents in order to coordinate processes. Principle approaches include the use of utility theory and game theory to represent rational choice, symbolic models of agent capabilities and roles, belief models, and graph models of organisational relationships.

5. How to recognise and reconcile disparate viewpoints and conflicting intentions among a collection of agents trying to coordinate their actions. Main approaches include assumption surfacing using automated truth maintenance techniques, parallel falsification and microtheories, partial global planning, knowledgeable mediation, standardisation, and various approaches to negotiation.

6. How to engineer and construct practical DAI systems; how to design technology platforms and develop methodologies for DAI.

Most systems have been characterised by (Gasser, 1991):

- the use of common interagent semantics with at most one or two meta- or contextual levels

- a reliance upon correspondence theories of representation and belief

- global measures of coherence

- the individual agent as the unit of analysis and interaction

- dependence upon closed-system assumptions such as shared and global means of assessing coherent behaviour, some ultimate commensurability of knowledge, or some boundary on the system.

From a postmodern viewpoint, these problems must be addressed without adopting a modernist view of agent cognition. Gasser (1991) suggests that several principles ought to underly the scientific and conceptual foundations of DAI systems from a social perspective:

1. AI research must set its foundations in ways that treat the existence and interaction of multiple actors as a fundamental principle. This raises the question of how representation, reasoning, problem solving, and action should be conceptualised from the social viewpoint. This requires a shift away from the focus of traditional AI upon the individual actor as the locus of reasoning and knowledge, and the individual proposition as the object of truth and knowing. Many of the concepts that have been basic to AI research (such as problems, knowledge, and facts) are regarded from the social perspective as reifications constructed through joint courses of action.

2. DAI theory and practice must address the basic tension between the local, situated, and pragmatic character of knowledge and action, and ways in which knowledge and action necessarily implicate multiple contexts. The meaning of a specific message is played out as a set of specific response behaviours; however, the responses may be local or distant along some dimension of distribution. Generality of knowledge requires that it should be transportable across contexts, but it must be possible to reintegrate knowledge into a local and situated context for use.

3. Shared knowledge is not a matter of several agents knowing the same fact interpreted in

the same way, but is a matter of aligning activities in a coherent way. This means that conflicts amount to conflicting actions rather than logical inconsistencies. The question is one of how mutually aligned and supportive commitments can occur and persist.

4. DAI theory and practice must account for resource-limited activity. Resource allocations are the product of the interactions among agents, and resources serve as a channel for interaction among agents.

5. DAI theory and practice must provide accounts of and mechanisms for handling the problems of *joint qualification* (how to establish a basis for joint actions, given the impossibility of fully specifying the assumptions behind a characterisation of any s i t u a t i o n ), *representation incommensurability*, and *failure indeterminacy* (identifying the source of, or reasons for, a failure).

6. DAI theory and practice must account for how aggregates of agents can achieve joint courses of action that are robust and continuable despite indeterminate faults, inconsistency, etc. which may occur at any level of the system.

All current approaches to distributed coordination rely on a global perspective at some level (eg. semantics, or communication protocols), and assume that the context of negotiation cannot itself be negotiated.

## A Discourse System for Agent Interaction

Any implementation of a distributed artificial intelligence (DAI) or multi-agent system (MAS) must address issues of how to provide the communication channels between system components. A number of tool sets extend basic terrestrial communications services to provide additional DAI/MAS facilities, generally emphasising the provision of platform-independent interagent communications and generic facilities implementing various control models, message routing schemes, task distribution schemes, memory management functions, and planning facilities. Examples of

such toolsets include OIS Semantics (Hewitt, 1991), SOCIAL, MACE, ABE, Agora, Cronus, Contract Net (CNET) (Adler, 1992).

Multi-agent approaches have been used both within the structure of individual spacecraft control systems, and as a model for systems having a number of internetworked (semi-)autonomous components. The agent metaphor is highly appropriate for use in the design and implementation of autonomous functions in space and ground support systems, since these systems already involve distributed, interacting agents in the form of human user, operational staff, and mission/spacecraft experts and specialists.

SOCIAL has been used in a prototype distributed system for decision support of ground operations for NASAs space shuttle fleeet (Adler, 1992). *SAGES* (Satellite Autonomy Generic Expert System) is a Rockwell project intended to support the reallocation of some ground segment functions onto the spacecraft, where primary functions (such as planning, scheduling, execution, and analysis) are identified with artificial agent roles within the on-board control system (Raslavicius et al, 1989). UNICORN is a blackboard-based multi-agent prototype for spacecraft autonomy, developed at General Electric, in which functions for fault diagnosis and related mission management operations have been developed, along with quantitative subsystem and environment simulations (Rossomando, 1992). Grant (1992/1992) describes a multi-agent approach to the design of the Columbus User Support Organisation (USO), based upon a Message-Based Architecture (MBA) Testbed; implemented in Smalltalk, MBA combines object-oriented constructs (classes, instances, attributes, methods, and messages) with forward-chaining expert systems and generative knowledge-based planning techniques.

This paper is not concerned with the provision of interagent communication channels within the satellite network, or with the application of a multi-agent metaphor within the architecture of a single satellite. Rather, it is concerned with developing a general model of agent interaction

based upon the postmodern conception of knowledge and intelligence described above.

Interagent cooperation mechanisms require several levels of linguistic competency:

- *transport services* must be present to provide point-to-point connections between agents in the network. For the satellite system, many of these connections are dynamic, being established and disconnected as allowed by the changing topology of the network. For any given network node, transport layer connections may be represented by *virtual channel interfaces*. The creation, maintenance, and disconnection of virtual channels is not considered here.

- particular protocols must be supported. If an agent can participate in message exchange facilitated by a particular protocol, it may be said to have the *exchange competency* required for that protocol.

- agents must be capable of engaging in negotiation to establish a joint activity, during which decisions are made about the participation/non-participation of particular agents, the respective roles of the participants, and the allocation of resources controlled by the participants. An agent capable of engaging in a particular form of negotiation may be said to have the *discourse competency* required for that form of negotiation.

- agents must be able to implement the operations constituting the *semantics* of message exchange; ie. the operations that implement the joint activities established by negotiation. An agent capable of implementing the operations required for a particular role in a particular type of joint activity may be said to have the *behavioural competency* required for that role in that type of joint activity. Behavioural competencies include the basic user-service functions of satellites as individual agents within the system.

An agent may participate in joint activities of a particular type if it shares a communication medium with other participants, and if it has exchange, discourse, and behavioural competencies required to participate in that type of activity. A type of joint activity and its associated competencies can be referred to as a *subculture*; an agent having competencies required to participate in a subculture can be said to be a *member* of that subculture.

Discourse Control

The mechanisms implementing particular competencies can themselves be treated as negotiable conventions. For example, protocol and discourse management procedures can be represented in a common formal language and distributed to all agents that are to participate in the corresponding subculture. In the DAI Open Information Systems (OIS) described by Hewitt (1991), deduction and representation-based reasoning processes are regarded as *microtheories*, based upon a closed world assumption so that derivations can be checked algorithmically for correctness without having to make any observations of the real world or consult any external information sources. Linguistic competencies can be developed and distributed as microtheories. A particular discursive activity may involve the elaboration of a temporary microtheory in the social sphere, with inferential procedures operating upon that microtheory as a social norm of discourse. Microtheories have important strengths in portability (ie. they can be described as stable inscriptions that can be easily stored, moved, and copied) and the self-contained decidability of derivational correctness.

An agent may be a member of several subcultures if it has the competencies required for those subcultures, for each subculture it may participate in a number of joint activities, and for each joint activity it may be in communication with a number of other participants. Apart from the requirement for the presence of appropriate basic competencies, agent participation in joint activities will be constrained by the availability of input and output communication channels, by the availability of on-board memory for the storage of discourse state definitions, and by the availability of sufficient power to support discourse processing and/or the activation of additional memory and communication channel

resources. The allocation of finite resources to discourse functions must take into account competing resource demands and their relative priorities. Resource demands will most likely vary according to the type of a joint activity, the number of its participants, the roles of its participants, and the state of discourse. Discourse control is therefore specific to particular types of joint activities and roles.

Here the particular example of distributed search and retrieval will be considered. This is a highly desirable set of functions for users of satellite bulletin-board and information services, particularly when system resources can be made available while hiding the (dynamic) network structure.

## Distributed Search and Retrieval

Search and retrieval is assumed to take place within the scope of a number of *source documents*. The source documents are each subdivided into *logical text units* (or LTUs), that are the individual targets of retrieval. The novel Distributed Search and Retrieval (DSR) system described here is intended to operate within a client-server environment characterised by:

- access via user interface functions having the following characteristics:

  1. A user will be able to request descriptions of search target types.

  2. search target types will be described to the user in terms of taxonomical categories and associated attributes.

  3. A user will be able to request a list of objects that conform to a particular profile specified according to category and attribute values.

  4. A list of item names will be returned to the user.

  5. The user may select any item on the list, and that item will be retrieved and presented on the interface display.

6. Depending upon the problem-solving context, the satellite network structure may not be visible to the user.

- low-level protocols provide Transport services.

- DSR Client processes provide a general purpose interface between DSR User Interfaces and DSR Search processes. DSR Client processes will generally be located at fixed ground nodes within the network.

- search and retrieval processes are implemented within DSR Servers located on physical network nodes together with their associated document files, a description of local taxonomies with their attributes, and a description of the documents or items belonging to each taxonomical category together with their attribute values. DSR Servers may be located either at ground nodes of the network or within satellites.

- DSR Searcher processes dynamically connect DSR Clients and other DSR Searchers to DSR Servers via the (changing) communications network. The search processes may be located on-board satellites or on ground nodes.

The overall architecture is shown on Figure 1. The DSR Client, the DSR Server, and the DSR Searcher are multi-user services, operating as continuous server processes.

On-board the satellite, the DSR Server is a system resource to be controlled. An on-board DSR Searcher must decide:

- whether to support a request for access to its associated DSR Server

- whether and how to pass DSR search requests on to neighbouring nodes in the network

- whether and how to pass results to a DSR Searcher on another node, irrespectively of whether the results come from its own server or that of another satellite

These control decisions must be integrated within the overall control of the satellite.

## Search Scoping Techniques

The issue of *search scoping* includes the issues of how to find nodes (and how many nodes) in a network during search (ie. the issue of *network span*), and what objects to search over at any given node (according to the *search context*). Search scoping is implemented using message and database metainformation.

To determine the *network span* of a search, the DSR Client will attach origin and message identification information, a message output time (ie. a time stamp for when the message is issued), and a timeout specification to each message that it issues. Upon receipt of a message, a DSR Server will first compare the origin and message identifiers of the incoming message with a record of the source and origin

dynamics to calculate how the model will change during a DSR session. This information might then be used by a routing strategy to provide return paths for information that may differ significantly from the original search message paths. However, as the network model increases in complexity, each satellite is increasingly likely to have circumspection problems, problems in keeping the model up to date, and greatly increased on-board computational loads.

A much simpler "reactive" solution is proposed here: message meta-information can include the virtual channel identifiers of the immediate connected neighbours of a given satellite that are the sources or sinks of currently active messages. Network dynamism is reflected at each node by changes in the active set of virtual channels presented to the control system. Any
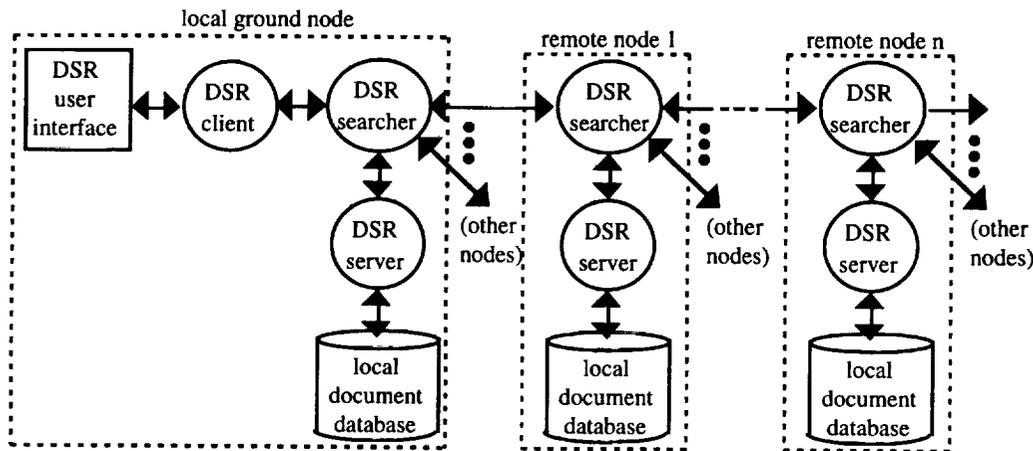


Figure 1. DSR Architecture.

identifiers of messages that it has already processed (stored for their timeout period). If the message is found to have already been processed, it will be ignored. This will eliminate cycles in the network search process. Similarly, the message timeout will be compared with the current system time, and if the timeout period has elapsed the message will be ignored.

Search scoping is critically affected by network dynamism. A sophisticated approach for LEO satellite constellations might be for each participating satellite (ie. agent) to have an up-to-date model of the connection state of the network, and to use knowledge of satellite

message destined for a virtual channel that no longer exists is deleted. This is a very simple scheme (a "flooding" protocol) that results in a DSR scope within a LEO network equivalent to the subnet that exists for long enough for the set of bidirectional message exchanges required by the protocol to be completed, where each DSR Client-server exchange in the session is achieved along a single physical route. Within a satellite constellation containing GEO satellites, the reactive system has a high potential to achieve full network coverage (if message timeout parameters are set high enough). For example, a rule stating that each successive node in a path must belong to a different segment of the

constellation (ie. LEO or GEO) will allow the system to use the stable configuration of GEO satellites to overcome limitations arising from the unstable subnet of LEO satellites. Virtual channel interfaces between LEO satellites and GEO satellites will be highly stable, lasting for one third or more of a LEO orbit period, with smooth and lengthy changeover periods between GEO nodes.[1]

The issue of *search context* is the issue of how to improve the relevance of retrieved objects according to the context of the search. Context-dependent searching is implemented using LTU categorisation and search filtering by category and feature constraints. Taxonomical information can be regarded as a form of *self-description* passed between agents, or as a description of the resources controlled by particular agents. In any case, an agent using taxonomical information must be capable of processing that information in order to use it in the specification of a search context. The ability of client processes to process and use those descriptions is critical in the creation of a coherent "society" of computational agents, and requires a higher level of standardisation than that involved in the definition of the DSR protocol described here.

It is possible to classify a single LTU by more than one taxonomy, and all LTUs must be classified by at least one taxonomy. Taxonomies are structures representing metainformation about LTUs, such as position within large-scale text structures (such as traditional books), subject matter, and purpose. It is possible to specify logical conjunctions of categories such that all objects retrieved during a particular transaction will belong to the subset of objects defined by the particular logical expression expressing the logical scope (or document subset) of that search. The specification of constraints upon attributes supports more specific forms of filtering. An abstract taxonomy corresponds to a schema that models object types, object interrelationship types, object attribute types, and subtype/supertype relationships between object types. The

classification and description of a set of LTUs using a taxonomy corresponds to the population of a database with a particular model of instances of the types described by the schema.

## A Distributed Search and Retrieval Protocol

Search and retrieval consists of three distinct types of transaction: getting taxonomy descriptions, getting taxonomy items, and getting a selected LTU. This section describes in more detail the message exchange between DSR agents associated with each of these transaction types.

## Get Taxonomy Description

### DSR Client

- receive a request for local DSR Taxonomy descriptions from the DSR User Interface process
- send a request for local DSR Taxonomy descriptions to the DSR Searcher
- receive a set of local DSR Taxonomy descriptions from the DSR Searcher
- send the set of Taxonomy descriptions to the DSR User Interface process

### DSR Server

- receive a request for the local DSR Taxonomy description from the DSR Searcher
- retrieve the local DSR Taxonomy description
- send the local DSR Taxonomy description to the DSR Searcher

### DSR Searcher

- receive a request from a DSR Client or Searcher for DSR Taxonomy descriptions.
- send the Taxonomy description request to the local DSR Server
- send the Taxonomy description request to DSR Searchers located at all neighbour nodes other than the immediately preceding node in the path traversed by the request
- receive a Taxonomy description from a DSR Server or Searcher
- send the Taxonomy description to the specified DSR Client or Searcher

---

[1] A detailed analysis of the performance of different network configurations and different reactive routing rules is beyond the scope of this work.

## Get Taxonomy Items

*DSR Client*

- receive a request for a list of items (LTUs) belonging to a specified taxonomical category, or a set of categories, from the DSR User Interface process
- send the request for a list of local items belonging to a specified taxonomical category, or a set of categories, to the DSR Searcher
- receive the list of items belonging to the specified specified taxonomical category, or a set of categories, from the DSR Searcher
- send the list of items to the DSR User Interface process

*DSR Server*

- receive a request from the DSR Searcher for a list of items (LTUs) belonging to a local DSR Taxonomy category
- retrieve the list of items belonging to the specified category
- send the list of items to the DSR Searcher

*DSR Searcher*

- receive a request from a DSR Client or Searcher for a list of items belonging to a DSR Taxonomy category of a specified DSR Server
- send the list request to the specified DSR Server or the next Searchers en route
- receive a list of items belonging to a local DSR Taxonomy category from a DSR Server or Searcher
- send the list to the specified DSR Client or Searcher

## Get LTU

*DSR Client*

- receive an LTU specification from the DSR User Interface process
- send request for LTU to the DSR Searcher
- receive specified LTU from the DSR Searcher
- send the LTU to the DSR User Interface process

*DSR Server*

- receive request for LTU from the DSR Searcher
- retrieve the specified LTU
- send the LTU to the DSR Searcher

*DSR Searcher*

- receive an LTU retrieval command from a DSR Client or Searcher
- send the retrieval command to the specified DSR Server or the next Searcher en route
- receive an LTU from a DSR Server or Searcher
- send the LTU to the specified DSR Client or Searcher

## Message Routing and Scope of Joint Activity

All message transmission associated with this protocol must satisfy the conditions that: the current time from origin for any given message is less than a timeout period specified within the message header, the next virtual channel that a message is destined for must be currently active, and the next virtual channel that a message is destined for must not be its immediate source channel. If either of the first two of these conditions is not satisfied, then a message is deleted instead of being transmitted.

Message routes define the scope of a joint activity, and it is within this scope that a searcher agent decides whether or not to participate in a joint activity. However, the network topology may change during the course of a joint activity. Searcher agents involved in Get Taxonomy Description transactions will define the scope of a joint activity by sending requests to all neighbouring nodes that satisfy the above three conditions. The return paths for these messages define the scope within which local taxonomical descriptions are known to be valid. Between the completion of these transactions and the beginning of new transactions based upon their results (in particular, Get Taxonomy Items requests), some satellites may leave the network while others may join it, depending upon timeout parameters, link bandwidths (ie. data rates), and orbital characteristics. This means that, unless a

transaction completes prior to any such changes, one of the following tactics must be employed:

- taxonomical item retrieval may be initiated with dynamic scoping. Previously acquired taxonomical data may include items that are no longer available, and items may now be available that are not "known" to the initiating DSR Client. Nevertheless, new agents entering the network may be included in retrieval operations.

- new agents are excluded from retrieval operations by using path information to directly address specific DSR Servers. This approach has the advantage of reducing system bandwidth usage, but has the disadvantages that some retrieval operations will not succeed due to agents having left the subnet, and potentially usable information from new agents that have recently entered the potential scope of the joint activity will not be accessible.

A similar tradeoff occurs in relation to the retrieval of a specific LTU: if the LTU is retrieved by explicit route information, network bandwidth is conserved; however, if the network topology changes, the item will not be retrievable, even though it may still be within a dynamically defined scope. Explicit routing is assumed here.

### Integration with the Behavioural Spacecraft Control System

The following specifications summarise general discourse rules:

- an agent may be a member of several subcultures

- for each subculture it may participate in a number of joint activities

- for each joint activity it may be in communication with a number of other participants

- agent participation in joint activities will be constrained by:

- the availability of input and output communication channels

- the availability of on-board memory for the storage of discourse state definitions,

- the availability of sufficient power to support discourse processing and/or the activation of additional memory and communication channel resources

- the allocation of resources to discourse functions must take into account competing resource demands and their relative priorities. Resource demands will most likely vary according to the type of a joint activity, the number of its participants, the roles of its participants, and the state of discourse (eg. the negotiation process may require less power that the execution of operations required by an agent once it is committed to participate in a joint activity in a particular role).

- the current time from origin for any given message must be less than a timeout period specified within the message header, or else the message will be deleted

- the next virtual channel that a message is destined for must be currently active, or else the message will be deleted

- a message cannot be sent to its immediate source channel

These rules must be implemented within the behavioural control system of the spacecraft. It is necessary to support multiple joint activities, roles, and communication links, and to make decisions about resource allocations to requests arising from negotiation processes and lower level survival and service competencies. This can be done using an arbitration mechanism that can assess relative priorities between all of these requests. In general this represents an elaboration of the emergent planning and scheduling system described by Lindley (1994) in which all requests for resources, including those arising from negotiation, are regarded as competing goals. This system is not described in detail here.

## Conclusion

This paper has described a distributed search and retrieval (DSR) system for open satellite constellations. The mechanisms for language interaction and negotiation are regarded as social artefacts, with agent behaviours driving the use and application of the language system. Resource allocation to joint activities is achieved by a behavioural control system, and this and the generation of appropriate action sequences (ie. "planning") is achieved in an emergent, non-deliberative, and decentralised way. The system constitutes a distributed multi-agent system that relies upon metaknowledge within the DSR environment to guide the search process and provide search filtering according to the problem solving context. The DSR system is typical of most current multi-agent systems in requiring a priori common objects such as interaction languages, metaconcepts, and behavioural rules or programs to ensure that agents conform to standards. Multi-agent systems as models of human societies require a model of the standard formation process itself, rather than particular standards, to account for the ongoing process of aggregation (as a process, rather than a state) and the fluidity of aggregate boundaries (in terms of knowledge and action) (Gasser, 1993). More fully social computational systems should have the capacity to generate, modify, and codify their own local interaction languages, have degrees of structure and reification that increase and decrease with use, and modify their knowledge and activity structures at all levels of analysis (ie. communities of programs should evolve the languages in which they are written). However, this is not necessary for the successful engineering of systems that depend upon cooperation to achieve openness and flexibility in their functionality, and in this case standards are a necessary prerequisite for the integration of systems into a common facility.

## References

Adler R. M. 1992 "Coordinating Complex Problem-Solving Among Distributed Intelligent Agents", *Telematics and Informatics*, 9(3/4), 191-204.

ASRI 1994 *Australis-1 Top-Level Design Document*, Australian Space Research Institute, Sydney, Australia.

Brooks R. A. 1986 "A Robust Layered Control System For A Mobile Robot", *IEEE Journal of Robotics and Automation*, RA-2, no 1, 14-23, March.

Gammack J. G. and Anderson A. 1990 "Constructive interaction in knowledge engineering", *Expert Systems*, Vol. 7, no. 1, Feb..

Gasser L. 1991 "Social conceptions of knowledge and action: DAI foundations and open systems semantics", *Artificial Intelligence*, Vol. 47, 107-138.

Gasser L. 1993 "Social Knowledge and Social Action: Heterogeneity in Practice", *Proceedings, 13th International Joint Conference on Artificial Intelligence*, Morgan Kaufman Pubs. Inc., 751 - 757.

Grant T. J. 1991/1992 "A review of Multi-Agent Systems techniques, with application to Columbus User Support Organisation", *FGCS7*, 413-437.

Hewitt C. 1991 "Open Information Systems Semantics for Distributed Artificial Intelligence", *Artificial Intelligence*, 47, 79-106.

Kirsch D. 1991 "Foundations of AI: the big issues", *Artificial Intelligence*, V. 49, 3-30.

Lindley C. A. 1993a "An Autonomous Satellite Architecture Integrating Deliberative Reasoning and Behavioural Intelligence", *Telematics and Informatics*, 10 (3).

Lindley C. A. 1993b "A Behavioural System for Autonomous Spacecraft Power Distribution and Control", *Artificial Intelligence and Knowledge Based Systems for Space*, 4th Workshop, May 17th - 19th, ESTEC, Noordwijk, The Netherlands.

Lindley C. A. 1994 "On-Board Emergent Scheduling of Autonomous Spacecraft Payload Operations", *9th Goddard Conference on Space Applications of Artificial Intelligence*, May 10-

12, NASA Goddard Space Flight Centre, Greenbelt, MD, USA.

Maes P. 1993 "Behaviour-Based Artificial Intelligence", *Proceedings of the Second International Conference on Simulation of Adaptive Behaviour* , MIT Press.

Raslavicius L., Gathmann T. P., and Barry J. M. 1989 "An Artificial Intelligence Framework for Satellite Autonomy", *Proceedings, Second International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Vol 2, 536-543.

Rossomando P. J. 1992 "The Achievement of Spacecraft Autonomy Through The Thematic Application of Multiple Cooperating Intelligent Agents", *Telematics and Informatics*, 9 (3/4), 205-219.

Waugh P. 1992 *A Reader in Postmodernism*, Edward Arnold.