University of Cambridge
Cambridge Service Alliance

Executive Briefing

TOWARDS SELF-SERVING AIRCRAFT:

Revolutionising the service supply chain

Authors:

Alexandra Brintrup
Member IEEE

Duncan McFarlane
Damith Ranasinghe
Tomas Sanchez Lopez
University of Cambridge

Kenneth Owens
The Boeing Company

The Cambridge Service Alliance

The Cambridge Service Alliance is a unique global partnership between businesses and universities. It brings together the world's leading firms and academics, all of whom are devoted to delivering today the tools, education and insights needed for the complex service solutions of tomorrow.

About the Cambridge Service Alliance

Founded in 2010 by BAE Systems, IBM and the University of Cambridge's Institute for Manufacturing and Judge Business School, the Cambridge Service Alliance brings together world-leading organisations with an interest in complex service systems to:

- Conduct insightful, yet practical research to improve the design and deployment of high-performance complex service systems.
- Create and develop industrially applicable tools and techniques that deliver competitive advantage.
- Provide an unparalleled network of academics and industrialists that share experience, knowledge and insight in how better to design and deploy high performance complex service systems.
- Develop and deliver public and member-only education programmes to raise the skill levels of organisations.

Joining the Cambridge Service Alliance

Industrial members

The founding industrial members are BAE Systems and IBM. The Cambridge Service Alliance will bring together up to eight further companies prepared to make significant and long-term contributions to support the Alliance. Benefits of joining include:

- Challenging yet practical insights into the design and delivery of high-performance complex service solutions.
- Practical tools, techniques and methodologies.
- Education and training to enhance capabilities in service and support.
- A stimulating international network of the world's best talent engaged in solving problems associated with complex service solutions.

Academic members

The Alliance draws on members from across the University of Cambridge, initially from the Institute for Manufacturing and the Judge Business School.

Internationally leading researchers and educators will be invited to join the Cambridge Service Alliance to meet specific research requirements and the needs of industrial members.

Further information

Email: contact@cambridgeservicealliance.org
www.cambridgeservicealliance.org

Acknowledgements

Work Sponsor

The work reported in this executive briefing was conducted by colleagues of the Cambridge Service Alliance, working in the Distributed Information and Automation Laboratory (DIAL) at the Institute of Manufacturing, in collaboration with the Boeing Company.

Contents

Executive Summary

Imagine if a complex supply chain could organise and manage itself with comparatively little human intervention. Take the aircraft service supply chain, for example. What if the parts in an aircraft were able to detect when they were due for servicing and arrange their own replacement, in an efficient and competitively priced way? Or products in a supermarket were able to negotiate and arrange their own restocking and supply?

It may sound like something from the pages of science fiction novel, but at Cambridge University a team of researchers are developing the information systems to make this vision a reality. In doing so they are revolutionising the supply chain management and operations with significant implications for a wide range of industries.

Like many other organisations, companies in the airline industry are under intense pressure to reduce costs and increase performance. Increased fuel prices and tougher market competition, for example, are just two factors challenging profitability. One area that offers opportunities for performance gains and cost reduction is the hugely complex aircraft service supply chain, which involves the procurement of many thousands of parts, and the servicing or maintenance of products during their life in service.

Innovations in the field of intelligent objects technology, offer some radical solutions to the supply chain challenges that businesses face. These are objects – an aircraft part for example - that depending on their level of intelligence, are self-aware, have goals and decision making autonomy, and can take action to perform certain tasks.

Together, in collaboration with the Boeing Company, researchers at the University of Cambridge are taking the intelligent object concept into the service domain, creating the self-serving asset. This is an information based representation of a part or assembly that is uniquely identified, can communicate effectively with its environment, retain or store data about itself, deploys a language to display its features and requirements, and can participate in or make decisions relevant to its own destiny.

To be of value, self-serving assets need to be able to act tactically, operationally and strategically. Ideally they should be able to: monitor their environment in order to decide on service actions; decide on service needs and select providers to serve them; interact with providers and other assets to make cooperative service decisions; and monitor whether the anticipated service activity has taken place, re-ordering if necessary.

This vision of self-serving assets requires sophisticated multi-agent, sensor and identification technology. It is here that the University of Cambridge research team has made a critical breakthrough, creating a multi-agent system architecture, through which software agents can act on behalf of their physical counterparts.

The model the team developed incorporates software agents representing individual components and component communities on the demand side, as well as the suppliers. In addition, there are software agents that are tasked with searching for suppliers, and for resolving completion for resources through auctions.

The self-serving asset agent platform has the potential to deliver reduced complexity, reduced time to service, less risk of system failure, and better decision making. These developments translate into some very real benefits for airlines and stakeholder service providers. These include: reduced costs of computation and communication, reduced risk of central unit failure, reduced time spent on service procurement, better decision making, increased data accuracy, and more. The intention is that the asset would, effectively, "manage its own affairs" including maintenance scheduling, part replacement and condition monitoring.

There is still some way to go to fully achieve the team's vision. Once realised, however, the implications for other industries facing similar supply chain challenges are immense. In the shorter term, it is already clear that the self-serving asset is viable, and likely to revolutionise the aerospace service supply chain. ■

Reimagining supply chain operations

Organisations operate in an increasingly complex world. Take the aerospace industry, for example. Managing the aerospace service supply chain effectively and efficiently is an immense challenge. A Boeing 747-400, for example, contains some six million parts. Just consider some of the supply chain activities involved in manufacturing, supplying, maintaining and operating an aircraft like the 747-400 –or a fleet of aircraft. There is, for example, the procurement of many of the parts, the updating of software or human resources, the servicing or maintenance of products during their life in service, and that is just small part of it.

And think about the challenging environment in which these activities take place. There are unpredictable service requests to contend with; the frequent pull of multiple resources from multiple partners, and the problems associated with the dynamic search for resources based using multiple performance criteria. Much of this happens on a global basis, adding another dimension to the complications involved.

Coping with complexity costs money - lots of it. No-fault-found investigations cost the aerospace industry $300 million per year, for example. At the same time, while the complexity involved in managing global supply chains continues to increase, a number of market drivers point to a pressing need to reduce costs. In aerospace, for example, greater demand for flights, increasing fuel prices, and tougher market competition, means airlines are under considerable pressure to find ways to drive down costs and improve performance.

Imagine, though, if it were possible to create a system where aircraft parts could monitor their own condition, detect when it was time to order a replacement, order a replacement in a price efficient manner, and schedule a refit. It sounds like an idea suitable for the pages of a science fiction novel, yet the world of self-service assets – in this case aircraft parts - is fast becoming a reality.

The catalyst of this potentially industry changing technological breakthrough is the collaboration between the Boeing Company and the University of Cambridge, and in particular, their work on the development of intelligent objects, and their application in the service supply chain. These are objects that, depending on their level of intelligence, are self-aware, have goals and decision making autonomy, and can take action to perform certain tasks.

Boeing and Cambridge University are taking intelligent object technologies and combining these with software agent technologies to create the self-serving asset. To enable this, the team have developed a multi-agent system (MAS) architecture, which forms the information backbone allowing self-serving assets to operate in a commercial environment.

The self-serving asset

A self-serving asset is the product of intelligent object and a software architecture that allows that object to perform certain tasks. Together, the object and the architecture give rise to the self-serving asset. This is an information based representation of a part or assembly that has a unique identification, can communicate effectively with its environment, can retain or store data about itself, deploys a language to display its features and requirements, and can participate in or make decisions relevant to its own destiny.

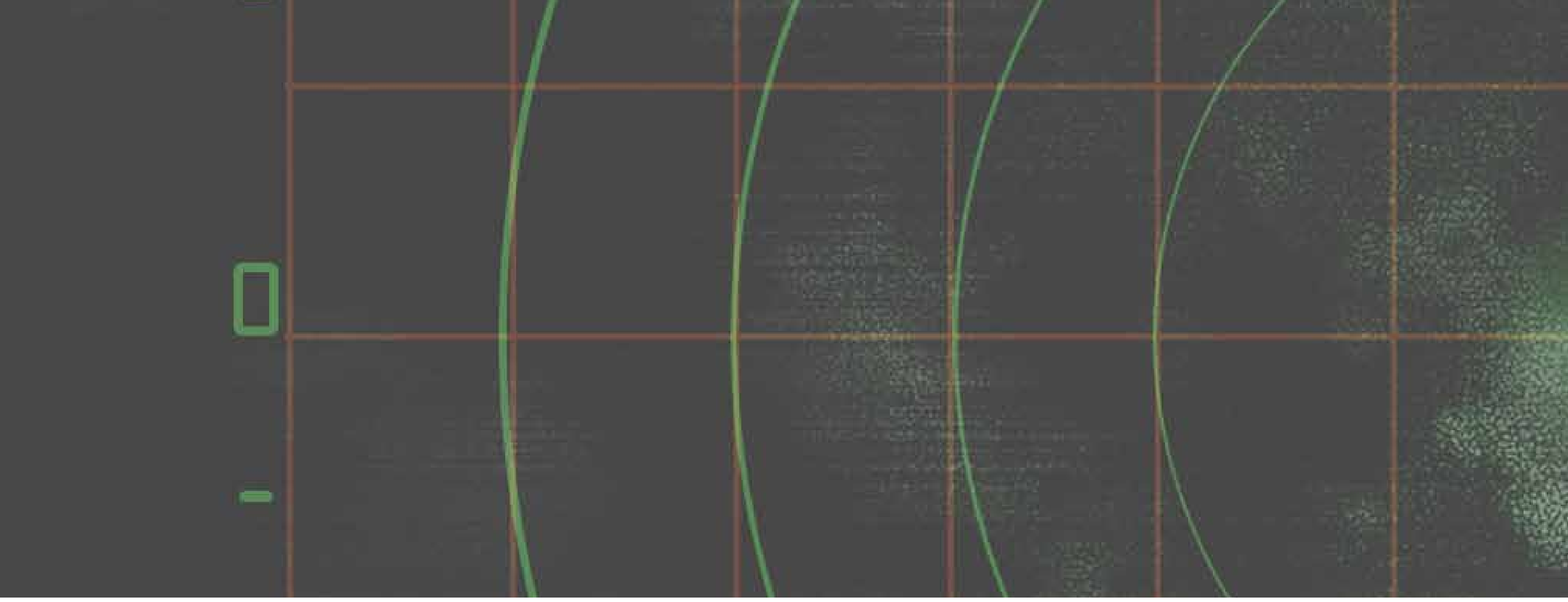
The information based representation is linked to its physical counterpart – a fuselage part or a lifejacket, for example – using identification technology, such as Radio Frequency Identification (RFID).

The self-serving asset has the potential to revolutionise the service supply chain. It can consider the needs of the stakeholders in the supply chain, satisfy them in an impartial fashion, and create a seamless, transparent environment where decisions are traceable, leading to a leaner service chain. In the case of aircraft, for example, common stakeholder needs include: minimising aircraft time on the ground while reducing service costs; increasing component life in service by accurate prognosis; and selecting best performing service providers.

At the same time deploying self-serving assets is likely to change the nature of the relationships - the rights and responsibilities - in the service supply chain. So, for example, maximising the self-serving asset’s life in service, allows an original equipment manufacturer to continue to create financial leverage through service contracts, long after the point of sale.

Requirements

To operate effectively a self-serving asset needs several distinct characteristics. For a start, it must be autonomous. It must be able to consider stakeholder goals, actively monitor its own needs, and



call for suitable actions to achieve goals in a persistent manner, all on its own. Activities such as asset condition monitoring, supplier selection, and part replacement calls need to be automated. At the moment, the majority of these tasks are performed by real people, rather than software agents.

Thus, ideally, and in more detail, a self-serving asset should: be self-aware, in terms of identity, location, health, expiry dates, and operation schedule. It should be capable of engaging in goal directed behaviour, maximising its life in service by autonomously deciding on its service needs, and managing the procurement of replacement parts, taking into account its resources and perceptions of its environment.

It should be able to act in the interests of different stakeholders. So it might select a supplier based on the previous performance of that supplier, or arrange a service schedule that minimises disruption to operations.

In order to fulfil its self-service function the asset will need to engage in communication with other assets or intelligent systems when searching and competing for resources. And, finally, it will need to be self-powering, throughout its lifetime, in order to perform its other functions, without relying on external power sources.

Benefits

If such a self-serving asset, and the systems required to support its effective operation, could be created it would provide numerous benefits.

Cost reduction would be significant. There would be more efficient use of resources, for example. Less time would be spent on service procurement. Tedious, often lengthy database searches and supplier conversations with be replaced with dedicated, consistent automated systems. Computer processing requirements are reduced, as are communication needs - one-to-one communication between assets and supplier resources means less centralised traffic.

Efficacy is improved. There is a reduced risk of central unit failure, for example. If distributed nodes fail to receive or update information, there is much less risk of system failure. It is only the nodes that are not working properly that remain out of the system.

Information gathering is improved, as information is collected dynamically, with assets continuously monitoring themselves and communicating with available resources. In its turn better more timely information means better decision making. The decision making mechanisms are formalised and automated leading to consistent and traceable decisions that take into account multiple criteria, and are based on structured methodologies.

Finally, another benefit of having self-serving assets of the type envisaged is that they increase the accuracy of the data. As it is the intelligent asset itself which stores, maintains, and serves as the sole source of its identity and state information, this greatly reduces the likelihood of data corruption.

The solution in practice

Fundamental to the creation and implementation of the self-serving asset concept is multi-agent technology, coupled with sensor and identification technology. The design and development of an efficient Multi-agent System in which software agents are able to interpret and act upon signals from individual parts, interact with other software agents and third party stakeholders in the supply chain, forms the information backbone of the physical asset on the enterprise network, providing the asset with self-aware, goal directed, and communicative behaviour.

Delivering an effective self-serving asset requires progress in Radio Frequency Identification and condition monitoring technology, alongside the embedding of decision making and competition resolving methodologies, together with the development of suitable information architectures.

This executive briefing details how the Boeing Company and a team from Cambridge University set out to meet this challenge, and to develop an appropriate Multi-agent System platform. It is worth noting that the Multi-agent System they have designed has the potential to revolutionise supply chain implementation in a range of industries – and not just the aerospace sector. ■

When smart objects have agents

Intelligent objects, smart objects or intelligent products, depending on what term you prefer, have been with us for some time. They are represented at various stages of the product lifecycle. In production, for example, there are objects that can steer through different levels of production by demanding processing from machines. In the logistics supply chain, objects monitor their own condition. While in the retail industry, objects dynamically reduce their price, or promote themselves to customers.

Levels of object intelligence

Objects may possess different levels of intelligence. So, for example, at a basic level of intelligence - Level 1 object intelligence –the object is not autonomous. There is unique identification of the object, but its state, whether that relates to its location, health, expiry, or production stage, is monitored by an external processing mechanism. Consequently, an external decision maker then considers the object's state and makes decisions about its future.

At Level 2 intelligence, an intelligent object will have goals, together with the decision making autonomy required to pursue those goals. So the object will be able to process data about its state and use this information to inform and enable actions that bring it closer to its goal.

Level 3 object intelligence adds the ability to interact with other objects, enabling it to engage in competitive or cooperative actions. The result is a self-managing and efficient planning and problem solving system.

Self-serving assets require a higher, Level 3 object intelligence. In particular, they need to be: uniquely identified - as they may have different service requirements; capable of monitoring their own status and the external environment, in order to decide on service actions; able to decide on service needs and select providers to serve them; possess the ability to interact with providers and other assets to make cooperative service decisions; and able to monitor whether the anticipated service activity has taken place, and re-order if necessary.

Level 3 intelligence through multi-agent systems

We speculated that the way to provide assets in the service supply chain with Level 3 intelligence was through multi-agent systems technology. Advanced software agent technology is able to provide objects with self-awareness and goal directed, autonomous behaviour. In a system with many assets, and where those agents have different goals, it can create the conditions for cooperative and competitive behaviour.

Multi-agent systems are able to replicate the situations with decentralised data, asynchronous decision making, and limited viewpoints due to impartial knowledge of individuals, that are typical of supply chains. Each stakeholder is able to participate in the process, with its own software agent negotiating on its behalf.

This concept has already been used in manufacturing supply chains, and in a more limited way, in the retail supply chain. In the service supply chain, information about the condition of an item is key to making prediction about its potential lifespan and need for replacement. Here, applications have been oriented towards systems that manage the health of an asset, particularly in military projects.

Previous studies have looked into integrating different aspects of the supply chain, such as production planning, strategic decision making, and logistics planning. Models created in the past, however, are at the enterprise level, with agents representing different stakeholders such as suppliers, production planners, and logistics planners. Furthermore, the models mostly consider the needs of the traditional, linear production supply chain, rather than the complex, unpredictable service supply chains such as those found in aerospace and some other industries. To use multi-agent systems and integrated object intelligence, in a way that best serves the aerospace service supply chain, new models are necessary.

Key requirements for self-serving asset

Constructing a multi-agent system for a self-serving asset means considering exactly what the asset needs to be able to do, and what are they key features that will deliver that functionality.

Certainly, to act effectively in the aerospace service supply chain, the self-serving asset needs to be both tactical (e.g. when planning servicing), operational (e.g. when executing plans), and strategic (e.g. when selecting suppliers). The asset also needs to be represented at an individual asset level, as each asset will have different requirements that emerge unpredictably. It may, for example, require service at different times, and might need different resources.

There are a number of factors which are particularly important

to consider when constructing the ideal multi-agent system. The system must make as little demand on computing resources as possible –it must be "lightweight". In its real world application there will be millions software agents representing parts and interacting with the system. The agents in the system must have fair divisions of responsibility in terms of computation and memory burden. For example a software agent that simply parses through supplier databases is not a good option as that would create a very large footprint, assuming there are multiple suppliers.

Another important requirement of the multi-agent system is that it must avoid creating information bottlenecks. There may be thousands of interactions taking place within a similar time frame. If the information conveyed in these interactions is directed through agents that are centralised nodes in the network, in a linear manner, this will inevitably create bottlenecks and lengthen processing time. Instead the system needs to have a distributed architecture where interactions can take place concurrently.

Finally, the way decisions are made in the system is very important. Decision making should be as efficient as possible, given that there will be competition among assets to acquire a scarce service resource, and among resources when multiple resources are available for use.

Here, efficiency refers to the speed of the decision making and is, therefore, connected to the elimination of bottlenecks. The competition arises when agents having conflicting utility functions or goals.

Decision making in the multi-agent system

A single software agent making a decision about the acquisition of scarce resources from multiple providers will need to do so quickly and efficiently. Scale this situation up to include thousands of agents interacting in the system and it is clear that way decisions are made is critical to the success of the system and the concept of the intelligent agent in this context.

The process is complicated by the numerous criteria to be considered in any decision making process. For example, if there is a choice to be made driven by competition for resources, the decision making criteria may include: location of the asset; urgency of maintenance; whether there are other assets that require maintenance in the vicinity (so that resources need to combine their travels and reduce associated cost); the price bid offered by the asset; the nature of the contract between the asset owner and the resource owner; and so on.

Equally when selecting a particular resource available from many providers the criteria may include: cost and location of resource; contractual priorities; inventory control approaches, like first in/first out, or last in/first out; the remaining useful life of the resource; and the historical trail on how satisfied previously serviced assets were from a particular resource supplier.

This last criterion - the historical trail – is a good example of the complexities involved. This "trail" might include information about a number of factors that contribute to customer satisfaction - the speed with which the resource has replied to the asset in need, for instance, or the performance of the asset after being serviced. Then there is the question of how available this trust-based trail should be to other assets. Should other assets see the dealings of a resource or organisation in the past, or should the information only be available to the organisations or assets that dealt with that resource?

It may also be necessary to make biased decisions, where certain suppliers, for example, are preferred over others.

The team used specialised decision making theory to deal with the complex decision making that needed to take place within system. This included, for example, multi-criteria decision making (MCDM) together with state of the art evolutionary multi-objective optimisation (EMO) tools to reduce the number of alternative options available.

How the model works

We decided that three criteria were particularly critical when designing the multi-agent system. The ideal system should be lightweight, place as little demand as possible on computing resources, minimise communication bottlenecks, and promote efficient decision making. These were the must-have criteria for the agent architecture supporting the self-serving asset.

After experimenting with a number of possible agent models, a preferred model was decided on. In this model a number of different types of software agent interact to ensure that parts are serviced or replaced as appropriate.

The first type of agent is the component agent. Each serviceable

plane part and item of equipment is uniquely identified and has its own intelligent software agent. The component agent checks the component condition database periodically, and may initiate the search for a service supplier depending on the information it obtains there.

The health of individual components is monitored via an Integrated Vehicle Health Management (IVHM) system. The IVHM uses a sensory information processing system that makes a prognosis about the remaining life of an asset, and then proposes suitable actions for increasing its remaining life, including part replacement alarms. An expiry date alarm produces a similar trigger for perishable objects such as oxygen generators. There is also a flight hour based expiry, where schedules stored in a central database are queried by component agents.

The next agent encountered is the community manager agent. Each distinct group, or community, of parts is represented its own software agent. So, for example, a community agent may be responsible for the landing gear community, or the life vest community, for example.

Imagine that a component agent, having received an alert, reports to its community that it needs replacing. The community manager will check with other components of the same type to determine whether any of them are close to needing to be replaced or repaired. If they are, then a batch order can be generated. The component agents in the community then check with the condition database, and if there is a need for a near term replacement, this information will be sent on to the community manager. As more components join, the batch order increases, reducing the costs. The manager agent checks a configuration database for any model updates.

A third type of software agent is now introduced to the process. The yellow page agent, as the name suggests, acts as the link to find potential suppliers - performing service discovery within communities of suppliers and warehouses holding or already purchased spares.

The yellow page agent interacts with a fourth kind of agent, the supplier agent. Each supplier has its own software agent that receives orders, decides whether it can fulfil them, and proposes a contract. The supplier agent may need to make decisions about which asset should get serviced first. Suppliers may also have multiple service roles allowing them to be part of different yellow page communities.

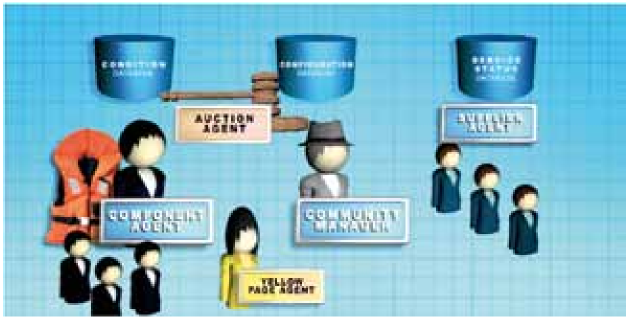
Once a supplier agent has responded, the community manager agent issues a preliminary contract and the supplier agent is able to accept the contract, or reject it. If a number of suppliers respond and are competing to supply the part, a fifth type of software agent can be deployed –the auction agent, which can run different types of auction depending on the situation.

Once any competition is resolved, on either the demand or supply side, a contract is exchanged between the supplier agent and the community manager agent. The community manager agent then sets up service alarms, based on the proposed service date in the contract. If the expected service time passes without the service taking place, the community manager agent sends an inquiry, and resolution activity takes place. This might include a prioritised service request and an update of the supplier reliability trail, to be used as part of future decision making about suppliers.

Self-serving asset agents form hierarchical groups that govern whether their behaviour is competitive or cooperative. Agents belonging to the same airplane's community do not have to be in competition with one another. They may, for example, come to a collective decision on the best time to request a service, reducing costs. On the other hand, agents of the same airline may prioritise who gets serviced first, through cooperative action, whereas agents from different airlines, depending whether or not there is a service agreement between them, may be in higher levels of competition and participate in an auction, in the case of a scarce, valuable spare part.

This, then, is the approach taken by the first self-serving asset demonstrator. Once we had conceived what we believed to be an effective multi-agent software architecture, to support our vision of a self-serving asset, the next step was to test it thoroughly. ■

Testing and performance



Simplified Agent Mode

A preliminary version of the agent architecture described above has already been implemented using the Cougaar agent development platform. Cougaar was developed through an eight-year research project, funded by the US Defense Advanced Research Projects Agency (DARPA). Its military logistics credentials, support for distribution and high scalability (tests included distribution of over 1000 medium weight agents across nearly 100 machines) were the reasons behind choosing Cougaar as our development platform.

The development version of the multi-agent system that we refer to in this briefing does not employ all the functionality described above, but does include: flight hour and time based asset expiry and health monitoring, batch orders, finding providers through service discovery, deciding on service providers using multiple criteria decision making tools, and the final contract exchange between the client and the provider.

Competition management through auctions, and post-service monitoring functionalities, are still under construction. For the purposes of performance testing we used life vests, fire extinguishers, and oxygen generators as example components. These are perishable goods for which the agents monitor flight hours and expiry dates.

A simplified simulation was run for the selection of these components using our multi agent model. The three main selection criteria were: the cost of a component quoted in the supplier proposal; the ease of reach of a service location, derived through the supplier location and that of the aircraft at a given time; and the contractual priority given to the supplier.

To evaluate performance we used four metrics. Response time examined the time the user has to wait until the agents have resolved a request, and how this changes as the number of agents and network nodes increases. Stability looked at changes in the number of messages exchanged – as the behaviour of the entire system could be affected if the number of messages grows quickly, or if these messages are not correctly managed.

The scalability of the model is deduced from the change in the number of messages exchanged or resolution time, as a result of increasing the number of agents involved in the multi-agent system. Finally, overall optimality was gauged by the outcome of the service selection. Having calculate what we believed to be optimal outcomes in terms of total cost, location ease and contractual obligations (i.e. minimal total cost and effort for locating components, and prioritised important contracts) we tested against those.

Overall, performance under testing was very encouraging. The proposed self-serving asset agent software provided satisfactory results in stability, scalability and resolution time. During different experimental runs, the number of messages remained stable and dramatic increases in resolution time and message numbers as the number of agents increased were not observed, pointing to a stable architecture.

On overall optimality findings show that either the optimal or second optimal solution is chosen with 99% significance in the current scope. On average during 77.6% of the runs the optimal solution was chosen, while no runs resulted in suboptimal solutions.

More testing is needed, though. In particular, further experimentation is required with greater numbers of agents, and distributed across geographically distant locations. In addition, testing using a lot more than just the three supplier selection criteria we used initially, will make the solution more suitable to real life scenarios. ■

Towards intelligent supply chains

The Multi Agent System based simulation that was designed and tested by the team shows considerable promise. We automated the search for suppliers by the component and community manager agents and got consistent results.

In the aerospace service supply chain, the same process would typically be carried out manually involving various database searches, paper based proposal exchanges with providers, and selection possibly based on human judgement. With self-serving assets not only are both decision making and procurement faster, but decisions are now traceable, and based on algorithmic models, eliminating human errors in data gathering and decision making. This is a huge improvement.

More tests are planned, with greater functionality and increased numbers of agents. The next stages of development will include a hardware integrated demonstrator involving Radio Frequency Identification tagged parts, and a condition monitoring framework using a Wireless Sensor Network, at the University of Cambridge Distributed Automation and Information Laboratory. The additional functionality will include competition management, external triggers, and post-service monitoring.

The self-serving asset agent platform provides a lightweight, autonomous software architecture to support what is a radical industry changing vision of intelligent self-serving assets. So far the preliminary tests provide promising results with regards to traceable, optimised decision making, thus reducing the time and effort required in the current frame of operations. The intelligent self-serving asset, therefore, has the potential to deliver substantial benefits - reduced complexity, reduced time to service, less risk of system failure, and better decision making. Overall, this should add up to considerable cost savings through greater efficiency as well as improved performance across a range of metrics.

There is, however, a long journey ahead and many challenges to be met before our vision of a self-serving asset is fully realised. Potential challenges include gaining cultural acceptance of distributed control, dealing with the complexity of the business case, achieving common standards, developing of hardware and protocols that will link physical objects to their network instantiations, and obtaining approval of relevant hardware architectures from legislative bodies. And dealing with these will not be easy.

But, while there may be some way to go, it is already clear that self-serving assets are viable, will potentially revolutionise the aerospace service supply chain -as well as other complex supply chains, and are a potential game-changer in this and other industries. ■

UNIVERSITY OF CAMBRIDGE

Cambridge Service Alliance