SOFTWARE FOR DOMAIN EXPERTS

helping free thinkers work better

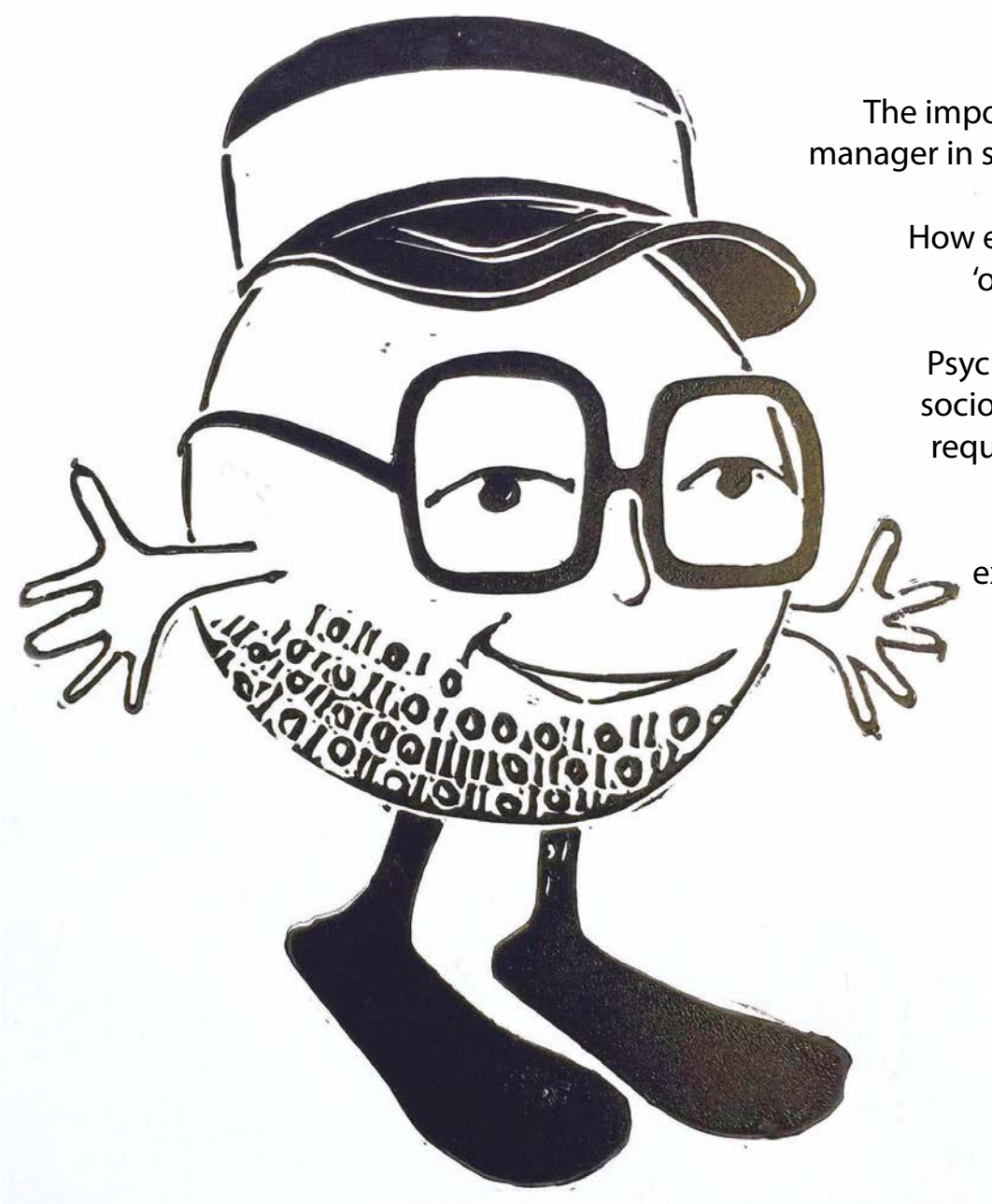
Special Conference Report

COMBINING EXPERTISE WITH SOFTWARE

Report from the fourth Software for Domain Experts Conference at The Cube, Athens, May 2 2017

Software which understands the customer

- How to build a software business for a small number of loyal customers - with few competitors
- The importance of the product manager in software development
- How e-learning can achieve 'outcomes' not 'outputs'
- Psychology, anthropology, sociology and linguistics in requirements engineering
- Applying security expertise to data systes
- Making the 'internet of things' work in a real business scenario
- Self aware computing



Sponsored by:

Ulysses SYSTEMS

SOFTWARE FOR DOMAIN EXPERTS
helping free thinkers work better

Connecting traditional expertise and software expertise in Athens

Software for Domain Experts held its 4th conference in Athens in May 2017 exploring the theme of connecting traditional expertise with software expertise and keeping everybody engaged

Topics included the importance of the 'product manager' role; upgrading e-learning from outputs (a certificate) to outcomes (where someone is better at their job); improving software requirements engineering; improving security of big data; making internet of things projects work; and self-aware computing.

By "domain expert" or "traditional expertise" we mean someone who is an expert in something other than digital technology or software – and uses that expertise as part of their work.

Dimitris Lyras, director of Software for Domain Experts (and also from shipping company Lyras Shipping and software company Ulysses Systems), introduced the conference by saying that today's software companies are largely looking for mass markets – but every domain expert is unique, and has unique needs for software.

But it is becoming easier and easier for software companies to make customised products, for an audience of a few or even one. It is also getting easier for individuals to build their own software to get the situation awareness they need for their work, using a variety of 'low code' or platform tools.

Today, the involvement of a domain expert in software is usually limited to selecting software packages or adjusting the final configuration, he said.

With better software tools, it should be possible for domain experts to achieve far more in their work – and correspondingly, there should be a market for this sort of software, and many business opportunities for companies to develop it, including for start-ups.

This is not a subject we hear much about in (for example) media discussions about software, which tend to focus more about the technology itself (as with the discussions on driverless cars or artificial intelligence), he said.

You could say that software has been developed until now basically taking advantage of the new technology which became available – the internet, data processing, storage, cell phones and everything else.

But we have reached a stage where it is possible for software to do far more than it currently does to help domain experts in their work, if people were willing to think about it more.

For software companies, it may prove much easier to build a sustainable business if you focus on a small market of domain experts, rather than chase large markets which look lucrative but have a small number of winners, Mr Lyras said.

For example, many start-up software companies in Greece are currently making tools to automatically scan CVs, and there may only be a final market for one or two companies.

But if you develop products for a specific domain, such as to help a certain sort of manufacturer monitor defects, you will probably find less competition, he said.

Because most software companies are hunting for large markets, there are many gaps which might be available for software companies happy with a smaller market, he said.

For example, very few large software companies are providing software tools for the 30 million refugees in the world. They may not make a market large enough for a giant software company to address, but it could be worthwhile for somebody, he said.

But there are many small software companies who would happily build something for a hundred thousand euros.

There are different ways which start-ups can be established – for example a small start-up affiliated with a larger company which provides resources.

This is a report from the fourth Software for DomainExperts Conference held at The Cube, 8 Kleisovis Str, Athens on May 2 2017

The full conference agenda, slides and videos of the talks are available online at
http://bit.ly/SFDEMay17
Some presentations and videos from the conference can be downloaded from the event website.

Report written by Karl Jeffery,
Editor,
Software for Domain Experts
jeffery@d-e-j.com
Tel 44 208 150 5292

Conference chaired by:
Dimitris Lyras
Layout by Laura Jones, Very Vermilion Ltd
Cover art by Alexandra Mckenzie

Software for Domain Experts Ltd.
c/o Future Energy Publishing,
39-41 North Road,
London, N7 9DP, UK
softwarefordomainexperts.com

The importance of the 'product manager' role

One of the best ways to ensure your company makes something which meets customers' needs might be to have a "product manager" in charge of each product, said Christos Lytras, managing partner of Greek innovation consultancy Hippocampus.io

Perhaps the best way to ensure software meets customer needs is to appoint a product manager, with the sole responsibility of achieving customer satisfaction, said Christos Lytras, managing partner of Greek innovation consultancy Hippocampus.io.

Christos Lytras, Managing partner of Hippocampus.io

The product manager keeps everybody co-ordinated – including the engineers, the various people involved in development and implementation of the product, and the various employees/ stakeholders of the 'customer' company.

And if the company has many different products, perhaps it should have dedicated product managers for each product, he said. "That's something in my mind which would have the highest probability of working."

The "customer", in corporate environments, can be many different people with different agendas, needs and wants, all trying to get the software development process in different directions, and the product manager needs to find the right path.

And a common scenario in companies is that many people are involved in a product but no-one is explicitly responsible and / or accountable for its success or failure, he said. So people can be more interested in ensuring that it does not interrupt some other agenda, than making it work.

And often, software companies are interested in selling products which will meet their sales targets, not necessarily solving the customer's pain.

A further challenge is that much of the focus on customer needs can be lost between the initial product design and the end product, where you end up with something very different, he said.

Mr Lytras sees some of his "product manager" ideas coming into practise in the financial services vertical with companies like N26. The company provides a non-interest bearing bank account, which can store money, and a credit card, to withdraw money. The business was designed with a very narrow scope, to be as simple as possible to use, easy as possible to set-up, and work well on mobile phones.

What the product manager can do

The "product manager" needs a combination of technical expertise, business / domain knowledge, and user understanding, to ensure that the software product meets the customer's needs.

The product manager should have a role in design, implementation and innovation of the product. They need to understand how the business vertical works.

They should be able to identify opportunities, assess them and contribute to developing the product which gives value to a customer.

They should have knowledge of software development, leadership, business modelling, and financials. They should be able to understand the markets the company is trying to address and develop a strategy, he said.

This person needs to understand product lifecycle management, user experience (UX) and development technologies.

The product manager needs to manage the whole product development lifecycle – from building the product to the development to the market assessment, support over its lifetime and shutdown.

This means the person needs ownership over the product – which in a corporate environment means overall profit and loss responsibility.

The product manager will probably also need political or change management skills, to overcome obstacles. There are usually far more people telling you something can't be done than encouraging you to do it, he said.

Mr Lytras personally saw the need for this many times, in his previous employment for 15 years in the financial services industry, being part of teams which deliver software, trying to integrate new software into the existing core banking systems.

So all together, the product manager has a very special bundle of personal and professional skills and characteristics. "It is very difficult to be able to identify all this knowledge and experience in a single human being," he said.

Domain expertise

The product manager should also have expertise of the domain the software is being used in, such as shipping or financial services. This probably means that the person needs to have actually worked in that industry.

"Unless these people come from the industry it is very difficult to have a point of view inside the industry, being able to identify the pain points, [so they can] come up with a solution to cater to needs," he said.

"In certain industries, like the one I work in [financial services], if you haven't worked in it, it is very difficult to understand how it works."

Connecting customers and engineers

An important role of the product manager is connecting the customer with the coders, who are creating the software.

If there is no close connection between customers and coders, it is too easy for the end product to be very different to anticipated, and nobody has noticed.

Within this role, the product manager can discourage coders from following their own ideas, which may lead to a product different to what the customer wants.

Traditional software development achieves this connection through formalised processes, with documents showing what everybody has agreed will be developed, and keeping it on track with various stages up to user acceptance testing.

Modern methodologies like Agile and Sprint also have techniques for keeping everybody on track over 5 to 10 day work cycles, he said.

Getting e-learning from "outputs" to "outcomes"

Too many e-learning programs focus on achieving specific 'outputs', such as certifying that someone knows a specific thing, rather than 'outcomes', where the person ends up better at their job, said Aris Kotsakis of consultancy RDC, formerly with the Greek Ministry of Finance - the Public Revenue Authority of Greece

Too many e-learning programs focus on achieving specific 'outputs', such as certifying that someone knows a specific thing, rather than 'outcomes' – where the person is able to do better at their job, said Aris Kotsakis, director, strategic change management, RDC Informatics (www.rdc.gr), former senior advisor to four secretary generals of the Public Revenue Authority-Greek Ministry of Finance.

To improve the situation requires an effort from everybody involved in e-learning, including employees, their instructors, their managers, the system operators and the HR managers, he said.

Above all, the experience needs to be pleasant, which means it needs to work well for the employee, he said.

Mr Kotsakis was been involved in the first – ever- Operational Excellence project in Greek Public Sector (as the founder & head of the project steering committee).He used Experiential Learning as a strategic enabler to improve operational effectiveness at the Greek Public Revenue Authority (10,000 employees) for 4 years (2012 to 2016).

RDC's Aris Kotsakis

The project rollout included innovative experiential learning workshops in process excellence, accountability excellence and cultural mind-set change (from outputs to outcomes). It covered a "critical mass" human capital spread, with key-employee and key-manager participants over all operational areas of the organisation (Tax Administration, Customs & State's Laboratory Administration).

To make it work, you might want to use blended learning (mixing online learning with traditional classroom learning), adaptive learning (learning is customised to the needs of an individual learner), run from the right software application, with domain expertise mixed in, he said.

Mr Kotsakis' team managed to improve learning in the Ministry of Finance, which was a very complex multi-stakeholder learning environment, in terms of its high levels of bureaucracy and complex digital systems, he said.

What good looks like

For good e-learning, there should be a "unique, personalised 360 degree interface" for each employee, covering all the different user/employee experience (UX/EX) and real world cases someone is working on.

It is still common for employees of big organisations to be doing multiple different e-learning courses, all with different software applications, at the same time.

The employee should feel rewarded in some way, ideally in the real world (not just a 'high score'). For example, if you develop the solution to a real problem as part of the training, your manager might invite you to present it to the company board. Or perhaps your solution could be recognised by your peers in the company.

The e-learning system should have a feedback mechanism, where employees provide ideas about how features can be improved, and say how happy the learning experience was. The employee's manager should also be able to contribute.

Many companies have 'customer relationship management' systems for their (external) customers – perhaps a similar system would be useful for internal customers (employees), he said.

Companies can transform themselves into 'learning organisations', aiming to foster a learning culture, and continually assessing their e-learning systems,

They can try "adaptive learning" and "micro learning", with programs tailor made to the student's needs, where you identify learning gaps and try to fill them.

You can use a range of different training tools, like collaboration tools, social learning and digital libraries.

You can have learning based on real experience – a simulation of being on the job, for example a simulation of repairing an engine in real time.

Connecting small businesses with experts online

The EU funded "Open iSME" project aims to make it easier to connect small businesses with technical experts who might be able to assist them, said Effie Siaini, president of Hellenic Association of Young Entrepreneurs

The EU funded "Open iSME" project aims to make it easier to connect small businesses with technical experts who might be able to assist them, said

Effie Siaini, president of Hellenic Association of Young Entrepreneurs.

It is a project co-funded by the CIP (Competiveness and Innovation programme) of the European Union, to help 'unlock the innovation capacity for smaller firms that aspire to grow'.

Effie Siaini, President of Hellenic Association of Young Entrepreneurs

Small enterprises can register on the platform and connect with researchers. The website is at http://www.open-isme.eu/

A second stage of the project will look at connecting small businesses with private sector organisations, she said.

The website has functionality for searching for relevant experts.

One customer is a company transporting drugs around Greece for dentists, and wanted very precise software to track exactly where the packages were at all time.

A second example is a company making instruments for seismic monitoring in the oil and gas industry.

Taking requirements engineering forward

One of the biggest challenges in software development is "requirements engineering" – describing how to build software which would meet customer needs. Nikitakos Nikitas, Professor at University of the Aegean, gave some ideas about how it can be improved

"Requirements engineering", describing how to build software which meets customer needs, is extremely difficult to get right, as you will understand if you have ever struggled to get software to do what you want.

It can be technically described as "the process of defining, documenting and maintaining requirements," within a software engineering or systems engineering project.

The specification needs to be very precise, with minimum ambiguity for developers.

In order to put together a good specification, it helps to have a good understanding of subjects like cognitive psychology, anthropology, sociology and linguistics, he said.

An understanding of cognitive psychology might help you understand the difficulties people may have in describing their needs in a way that can be built in code.

An understanding of anthropology will help you take a methodological approach to observing human activities, and so understand how computer systems might help them.

Sociology provides an understanding of the enormous political and cultural changes caused by computerisation.

And linguistics is important because requirements engineering is largely about communication, he said.

Documents

There are different requirements documents which are used by different people.

The "user requirements" document can be stated in natural language. This can be used by the customers, contract managers and software architects.

You can have a 'systems requirement' document showing detailed descriptions of the system services, as a contract between the client and software contractor.

You can have a 'software specification' – a detailed software description, which can serve as a basis for design or implementation, written for developers.

You can have a list of "functional requirements" showing what the systems should (or shouldn't) provide. They need to be complete (covering all services), consistent and unambiguous.

The "non-functional requirements" are constraints – such as operating speed, time to market, cost. They typically relate to the whole rather than individual system features.

These non-functional requirements can be put in 3 categories – product related (efficiency, reliability, portability, usability, performance, space); organisational related (delivery, implementation, standards) or external requirements, such as interoperability, legislative, privacy, safety, ethics.

Then there are domain requirements which need an understanding of the specific domain, for example tolerance level of landing gear on an aircraft, and fibre optic communications during winter.

Requirements process

The requirements management process can begin with an "inception" stage, gathering a basic understanding of the problem, setting up communication channels, identifying stakeholders, recognising multiple viewpoints. Here, identifying the stakeholders and engaging the stakeholders is the most important, so everybody collaborates. 90 per cent of the total effort is the communication, he said.

The next stage is "elicitation", identifying boundaries of the system and getting the right amount of technical detail, understanding what is wanted and what might be possible to deliver. The requirements can also change over time.

A next stage can be collaborative requirements gathering – putting groups together to discuss the functional development to achieve the required quality level.

The next stage can be "elaboration," where the software engineer expands and refines the idea, and develops a refined technical model of the software functions, features and constraints. This work involves developing 'use cases'. You end up with an analysis model which defines the function, information and behavioural aspects of the problem.

Then comes "negotiation", a task of reconciling conflicts between what the customer wants and what can be achieved, including ranking requirements, identifying risks, making guesses of development effort, perhaps eliminating or combining requirements. A lot of projects fail due to the negotiation. Iteration / trial and error

The elaboration and negotiation process should sift through the 'real' requirements from the less important ones, or when people ask for something they don't really need. It can also constrain the development work within the available budget.

Next comes "specification", a final document produced by the requirements engineer, which serves as a foundation for software engineering, formalises the requirements of the system.

There is a validation task is to check the results, that all software requirements are unambiguous, there are no errors, omissions and inconsistencies, the work products conform to required standards,

There is a subsequent 'requirement management' task, of tracking any changes to the requirements, giving requirements unique identifiers, and making sure they are implemented.

Security for big data – ENISA's view

The European Union is particularly concerned about security of big data, particularly when the data is generated by the so called Internet of Things (IoT) devices. It advises on how security can be improved through the European Union Agency for Network and Information Security (ENISA). Rossen Naydenov of ENISA explained further.

The European Union Agency for Network and Information Security (ENISA) is a body based in Greece, which advises companies and organisations on better ways to manage information security, and sometimes sets rules.

ENISA's Rossen Naydenov

This represents another link between the world of traditional expertise and the world of software expertise, because it is bringing security expertise (which is very different to conventional software expertise) together to the digital world.

ENISA is particularly concerned about the security of 'big data', particularly when the data is created by IoT devices, said Rossen Naydenov, officer in Network Information Security at ENISA.

In the big data world it is common to have a large amount of different devices generating different sorts of information at different speed, in different formats.

A common definition of big data is data which has a high volume, velocity, variety and veracity. "Veracity" means the accuracy of the data, or whether it has been tampered with by anyone before you received it, has it been physically secure.

Ensuring there has been no tampering of the data between source and customer is hard, he said. There may be a business opportunity to make devices which can deliver data in a way that you can prove it hasn't been tampered with, Mr Naydenov said.

Ensuring veracity can be very hard if your company is gathering data from many different customers, and not all the customers support your company's identification methods, he said. The same challenge applies any time you are gathering data from a variety of sources.

Another veracity challenge comes from the large number of new devices coming onto the market, which may not use the same data protocols as the devices they replace. This might mean that "the information you are getting is not what you are expecting to receive," he said.

Another security issue with big data is that there is currently only one major platform Apache Hadoop. As we see with Microsoft Windows, if one company has a large market share this also leads to security problems.

Devices

There are many new sensors coming onto the market, which generate and transmit data – but they also create their own security challenges, if you don't know if they are also sending the data somewhere else.

This is a particular concern for internet of things (IOT) devices related to personal use. "Securing and protecting information is very important," he said.

These sensors are increasingly being used for energy management, including smart meters, which transmit personal data about electricity consumption. This needs to be encrypted, he said.

There could be business opportunities for companies which can provide a way to prove that a distributed system is secure and the data has not been changed.

Privacy protection is very important. Many home devices are sending information through the internet, and it should all be encrypted.

Software companies should maybe think more about security during the software development process.

There was a case where ENISA was involved in blocking sales of a child's toy doll, which had a camera and microphone inside, and would respond to a child's actions, and also send data back to the company – but the data transmission and storage was not secure.

What ENISA does

ENISA is the European agency for information security, which is funded by the European Union. Its main building is in Heraklion, Crete, and there is an operational and research office in Athens.

It covers a number of industry sectors, with staff from all around Europe.

Particular domains mentioned on the ENISA website include internet infrastructure, smart grids, finance, health, critical information infrastructure, law enforcement, energy, 'smart' cars / cities / airports / homes industrial control systems

It can play a co-ordinating role, for example co-ordinating telecom companies across Europe.

It also sets standards for information securitysystems which critical sectors should adopt (for example in energy, finance, water supplies).

ENISA does analysis of cyber security issuesin different sectors and publishes recommendations. ENISA sometimes looks at security issues in specific industry sectors.

For example it recently looked at security of data associated with box container transportation, where the data has to go through different organisations as the container travels.

Nokia - making the 'internet of things' work

Zois Kokkonis, product manager with Nokia Networks Business, gave some perspectives on how to make the 'internet of things' (IOT) work in a business environment

Zois Kokkonis, product manager with Nokia Networks Business, gave some perspectives on how to make the 'internet of things' work in a business environment today.

Zois Kokkonis, Product Manager with Nokia Networks Business

So many devices are being given some kind of network connectivity. But making it work "is not just about technology, it is about adapting this technology to our needs," he said.

One of the biggest challenges with any IOT project is getting the data connectivity, with a big enough bandwidth (data communication speed), low latency (communication time), and coverage.

A second challenge is getting all the different systems to talk to each other – and here you need standards. "No company can do it alone. There needs to be agreements on how the different systems talk to each other," he said.

Security is also a challenge.

A further issue is incorporating domain knowledge - understanding the domain which the system is being implemented in. No company has all the knowledge which is needed. There are different solutions for different industry sectors, he said.

But there also needs to be a 'horizontal' approach to developing an infrastructure for these services, in order to scale in a way that can address the needs of this market.

Above the devices, there is a 'platform' layer, where the software runs, which manages all of the devices, which could all be from different manufacturers.

Then you have 'services' – including all the integration efforts, planning and deployment.

Nokia is envisioning an IOT network landscape, to which any vendor and any customer will be able to connect into, he said.

Cars

Nokia was involved in an internet of things demonstration project for cars, also involving network communications company T-Mobile, research company Fraunhofer, and tyre manufacturer and vehicle electronics company Continental.

For this project, Nokia used a 'mobile edge computing' solution, where some of the data processing is done next to the sensor, rather than all of the sensor data being sent to a computer system elsewhere for processing. This reduces the latency.

The project team explored the development of a "passing assistant", where two cars can communicate when one intends to overtake another. It also explored an automatic braking system, where one car will automatically brake if the car in front brakes. Clearly for both of these applications there cannot be long delays for data communications and processing.

Normal networks might have a latency (communications delay) of 100ms – but applications like this need much shorter latency, 10-20ms, he said.

The work proved that "partnership is key to having a successful result," he said, with all members of the team making a contribution.

Fast development cycles

IOT projects typically need faster development cycles than many companies are used to, he said.

A project team may be able to work much faster, if it can build and refine prototypes, try them out with customers, and continually improve the business model so that it works from the perspective of all partners.

Software companies often talk about 'devops' – combining IT development with IT operations, he said. The same idea can be applied to other technology developments.

If companies insist on non-disclosure agreements, partnership agreements, and lengthy up front discussions about what business model to pursue, that can really slow things down, he said.

Getting the customer 'interface' can be tricky, particularly if your customer is not sitting in front of screen, but (for example) driving a car.

Standards

The push for standards can come from different directions, including a large company driving a standard, or a group of companies who see that it is in their mutual interest to make components integrate.

However there can be many complexities, for example if one company sees it would have an advantage for having a technology linked to its current offering adopted more broadly as a standard.

John Kontos – self-aware computing

Self-aware computing – such as computer systems which can explain where their suggestions are coming from, or better understand why the customer is struggling, could be a big exciting software advance

John Kontos, emeritus professor of artificial intelligence, Athens University, talked about trends in so-called 'self-aware computing', when a computer system has some kind of sense of what is going on, such as how their outputs are generated and whether the systems are malfunctioning, or if the customer is struggling with something.

John Kontos, Professor of Artificial Intelligence, Athens University

Most present computer systems cannot actually generate explanations, they can only follow instructions for computing the output asked by user. So it is basically another layer of software that monitors this computing and stores the path followed in order to use it for answering "why" questions.

"We don't attempt to implement human-like consciousness with machines. We are only inspired by human consciousness," he said.

Therefore, "self-aware computing systems don't replicate biological systems - they monitor their own operation and try to work out what is going on," he said.

A basic structure for a self-awareness system could have three levels. A ground level for knowing what data is to be processed, an object level for choosing actions, and a meta level for monitoring and control of the actions, he said.

Efforts to implement self-aware systems have been going on for years – an early one was the 'explaining expert systems' ideas of over 40 years ago, which aimed to be a computer system which would act like a human expert and may considered as self-aware at a primitive level, e.g. the MYCIN system for medical diagnosis.

Mr. Kontos said that artificial intelligence might be better seen as a bundle of different technologies, rather than a single consolidated technology. This can include specific technologies for data mining, question answering, self-aware systems, pattern recognition, knowledge representation, automatic reasoning, deep learning, expert systems, information extraction, text mining, natural language processing, problem solving, intelligent agents, logic programming, machine learning, artificial neural networks, artificial vision, computational discovery, computational creativity. So "self-awareness" could be seen as one of the strands.

XAI

The US DARPA (Defense Advanced Research Projects Agency) is encouraging the development of what it calls "Explainable Artificial Intelligence" or "XAI", which is a form of self-aware computing.

In this case, the self-awareness is for the system to explain to the customer or user why it is making the suggestion it is making, rather than just presenting the output of a black box.

As an example, instead of saying that the computer thinks a certain image is of a cat, the system might say that it has identified that the image shows an object with fur, whiskers and claws, and a distinctive two ear feature on its head, which means that it is probabilistically highly likely to be a cat.

Trusting the software

As the complexity of software increases, the amount which programmers understand about them reduces. The software can be developed by large groups of people, often with poor communication between them, he said.

This can be very dangerous if the software is used in critical infrastructure like air traffic control, power stations, energy grids, aeroplanes and trains. It also means that decision makers are far less likely to trust what the system is suggesting, particularly if they don't understand how the suggestion was generated.

To develop self-aware systems might require a new kind of software engineering.

"This computer system is going wrong"

One form of self-aware computing which could prove very helpful is software which can tell you when and why it is failing, and what the customer should do, rather than quietly fail and leave the customer to find out afterwards (such as when a WannaCry virus is encrypting all of the files).

One audience member, who runs IT for a Greek shipping company, said that to have software which understands itself when it fails would be "very important. I don't care about the rest. Is it science fiction?" he asked.

Complexity

Usually, the more complex the software is, the harder it is for the software to have self-awareness.

The software industry typically spends a lot of its time on testing, because it is too complex for anyone to be sure what it would do in any situation without testing it.

The complexity is increased when you are using an ecosystem of different products made by different companies, as many of us are with our computers.

There would need to be a way to pass the self-awareness on from one piece of software to another.

Examples

Mr. Kontos gave several examples of self-aware computing.

A drone could be programmed to avoid flying into people, trees, buildings and power lines.

A tractor could be programmed to run autonomously, sense sense obstacles in its path and stop, until the operator assigns a new path or says it is OK to proceed. It will also stop if it loses its (probably GPS) position data.

A system could make a recording of everything the user does –what the user could see on the screen, what the user did, and asked the question of users, so it keeps a formal representation of the entire interaction which can then be used to improve the user interaction.

There could be systems which monitor events, spots anomalies (where something deviates from expectations), tries to work out different strategies for dealing with them, and guides the user to follow one or more of them, while continuing to monitor. This technology will be necessary for autonomous sea-sailing vessels in the future.

Robotic spacecraft is another kind of system that need for self-awareness due to the fact that their control from the Earth may involve large time delays of the control and response signals due to the great distance between the spacecraft and earthbound command and control base. Future Mars and asteroid missions might be more successful if self-aware spacecraft is used.

SOFTWARE FOR DOMAIN EXPERTS
May 3, 2017, Athens Attendee List

Stefanos Capsaskis, Principal, 7L Capital Partners

Achilleas Choursoglou, Member of BOD, AMMITEC

Ioannis (John) Kontos, Professor, Athens University

Aristos Doxiadis, Big Pi Ventures

Artemis Maglara, Scientific Reseach & Development, CHOES Ltd

Stefanos Papandreou, Editor, Elnavi Monthly Shipping Review

Rossen Naydenov, project manager - security of the finance sector, ENISA

Karl Jeffery, Editor, Finding Petroleum / Digital Energy Journal

Tasos Makris, Information Systems Director, Gourdomichalis Maritime

Effie Siaini, President, Hellenic Association of Young Entrepreneurs

Christos Lytras, Managing Partner, Hippocampus.io

Charalampos Meletis, Independent researcher

Polizois Kokkonis, Product Manager, Nokia

Georgios Kasselakis, Partner, Openfund / Big Pi

Aris Kotsakis, Director Â– Strategic Change Management, RDC

Nikos Trampoulis, Owner - Developer, Red Pill IT Solutions

Alexander Kostopoulos, CEO, Reportbrain

Ioanna Malagardi, Consultant, Self employed

Dimitris Lyras, Director, Software for Domain Experts / Ulysses Systems

Natalie Novick, Startupcity.co

Panos Lagoutaris, CIO, Steelmet SA

Eleftheria Papadopoulou, Student

Dimitris Triantafillidis, Student

Yiannis Chatzinikolaou, Terrabyte

Kosmas Bougiouklis, consultant, Ulysses Systems

Eirini Fotiou, Business Development Manager, Ulysses Systems

Myrto Gkouzelou, Business engineer, Ulysses Systems

Raphael Lagos, Marine Consultant, Ulysses Systems

Angela Lyras, content analysis manager, Ulysses Systems

Evi Theocharidou, General manager, Ulysses Systems

Aggeliki Giouni, student, University of Piraeus

Dr Nikitas Nikitakos, Professor, Dept. of Shipping Trade and Transport, University of the Aegean

Ilias Tsamourgelis

What did you enjoy most about the event?

> **the original idea 2. the discussion after every presentation**
>
> *Tasos Makris, Gourdomichalis Maritime SA*

> **The very interesting subjects, the speakers, the exact following of the scheduled agenda.**
>
> *Achilleas Choursoglou, Association of Maritime Managers of Information Technology and Communications (AMMITEC)*