

TITLE: New PHP support for CICS TS

HOST: Hi, and welcome to the "Did you say Mainframe?" podcast series. This is where we regularly interview IBM technical experts who can help you to understand important IBM mainframe hardware and software issues. I'm your host Sherrie Abshire.

Today we're going to talk about CICS PHP SupportPac CALS that has recently been released from CICS.

Our guest today is Ian Mitchell from CICS Strategy in Hursley, UK. Ian, it's great to have you here.

SME: Hello there, it's very nice of you to invite me

HOST: Before we begin, I'd like to mention to our listeners that there is a SupportPac you can download. I'll be giving out more information at the end of this podcast.

HOST: [QUESTION 1] So Ian, this is yet another supportpac providing innovative features to the CICS TS version 3 customers.

SME: [ANSWER 1] Yes, and this one is very much complementary to the Atom Supportpac - CA8K - that we made available last year. Both of these are enablers for exposing valuable CICS services and data managed by CICS transaction to a wider variety of consumers. Both the Atom supportpac and this new one, known as CALS, both demonstrate the power of a simple approach to SOA, sometimes known as RESTful SOA or WOA - Web-Oriented Architectures. Atom is an example of a so called RESTful API and that supportpac provides a very easy way to allow consumers to discover feeds and access data directly from CICS. With our new PHP support made available just before Christmas, we're expanding the options for moulding and reusing CICS assets in an agile, Web 2.0 environment.

HOST: [Question 2] Could tell us more about what PHP is and what makes this technology a good match for CICS?

SME: [Answer 2] PHP is what is known as a scripting language. The term 'scripting' in this context means a language that focuses on ease of use, and speed of expression. That means that the language itself is easy to understand, natural for both experienced and less-experienced developers to write in, and provides many high-level built-in functions to achieve the results that the business is looking for. PHP has grown up in the open-source world and its creators have been highly focused on productivity and simplicity. PHP application today run a very high proportion of the websites around the world - that has been its incubator and driven a very fast evolution to the point where there are upwards of 3 million PHP developers around the world and something like one third of all websites have a PHP environment. With this history, PHP

2009 WebSphere System z Podcasts - 'Did you say Mainframe?' provides very attractive capabilities for a business looking for agility in their web applications, and we can carry over many of these qualities to the CICS environment. As a means of developing a RESTful interface to an existing CICS COBOL or PL/1 piece of business logic, a PHP script has many advantages.

HOST: [Question 3] IBM is offering WebSphere sMash with a PHP element, correct?

SME: [Answer 3] Yes, we see the power of a very dynamic and yet simple application environment being relevant to a number of situations. ProjectZero.org is the associated incubation project where the technology in WebSphere sMash evolves and the community can provide feedback, ask questions and steer the direction of WebSphere sMash. WebSphere sMash is a complete agile Web application platform for developing and running modern Web applications. As well as PHP, sMash provides the Groovy language which is an excellent path to the world of scripting for Java programmers.

HOST: [Question 4] You've mentioned RESTful a couple of times, can you remind us what that term means in a CICS context?

SME: [Answer 4] Certainly, it's actually a surprisingly simple concept - it's part of the umbrella term "Web 2.0". When I use the term in a CICS context, it's really shorthand for a number of common and simple techniques, with an underlying objective to aim for simplicity and wide applicability. The principles are to use common Web technologies - HTTP as a transport, URLs as a means to identify resources and services, and XML as a data format. CICS now has excellent support for these technologies and the guidance in the principles of REST - which stands for "REpresentational State Transfer" by the way - provide a framework for architecting simple interfaces to existing business services.

HOST: [Question 5] So RESTful services are part of SOA?

SME: [Answer 5] Oh, most definitely. Both Atom support and PHP are complementary to CICS' other SOA enabling technologies such as our SOAP Web services support. RESTful architectures apply in simpler situations than can be supported by SOAP Web services - for example you don't have the richness of support for WSDL service descriptions, the ability to manage services in a registry and benefit from standards such as WS-Atomic Transactions.

HOST: [Question 6] Returning to PHP as a technology for a moment, what does it offer that's different to, say, COBOL or Java?

SME: [Answer 6] If I were to try to sum that up in two words, I'd say "instant gratification"! The classic "Hello

2009 WebSphere System z Podcasts - 'Did you say Mainframe?' world" program in PHP is literally five words - a beginning and ending XML tag enclosing "echo", "hello" and "world". There is no compilation with PHP, so the script containing that program just needs to be placed in a directory on the server and a web request can be made to invoke that function. Scripts can be edited with any text editor and are normal text files that can be easily put into your source code management system.

HOST: [Question 7] OK, so you can certainly start simple - what else does PHP offer?

SME: [Answer 7] The heritage of PHP is to enrich web sites by allowing dynamic content to be generated. So some of the simplicity stems from introducing "programming" to people used to just creating static content. A PHP script can contain a mixture of information to be delivered to the service requester 'as is' - could be HTML, XHTML or XML - and PHP code. When a request is made, the static content is delivered and the PHP code executed to provide the dynamic content. This content is typically retrieved from databases or other backend services. You might compare this to a CICS Web application that uses the Document API, there being a similar mixture of 'boilerplate' content and business logic. PHP offers all the usual control structures (IF... THEN... ELSE...., SWITCH, FOR loops, etc) to express business logic. It also has a rich variety of data types but much of its power is in the ease with which strings and arrays are handled. Lately, objects and class structure have been added to the language making it much easier to make large PHP applications more modular.

HOST: [Question 8] So PHP is a modern language. You mentioned access to databases and backend services, how does that work?

SME: [Answer 8] Much of the heavy lifting in a PHP application will actually be handled in high-level library functions invoked from the PHP. These are typically implemented in C or another compiled language. Many library functions have been created as the PHP community has seen the need for efficiency in database access, XML parsing, image manipulation and other web application function. In offering the PHP Data Objects API to access DB2 data from a CICS PHP application, we have ensured that this is 100% integrated with the existing DB2 adapter for CICS, so that it benefits automatically from the performance and transactional qualities of service CICS customers expect. Although PHP script themselves are not compiled ahead of execution, they delegate a significant amount of function to libraries of optimised compiled code. Of course, modern concepts of just-in-time compilation are being applied to PHP itself as well. In the CICS supportpac we have included ports of many of the standard libraries.

HOST: [Question 9] I guess one of the unique aspects of PHP with CICS is the ability to access existing CICS applications.

SME [Answer 9] Absolutely. With PHP executing within the CICS environment, any existing CICS function is typically only an EXEC CICS LINK away. We've provide PHP APIs so that invoking existing business logic is a very efficient and natural thing for a PHP developer to do. We have also enabled access to DB2 data directly from a PHP script in CICS via the PHP Data Objects API. We have also provided some lightweight tools that enable classes to be created that perform the mapping of data between PHP data-types and a CICS application's commarea. In line with the philosophy of simplicity inbedded in PHP, we've tried to make the invocation of CICS business logic from a PHP script a very easy thing to do.

HOST: [Question 10] Can you tell us about any other notable aspects of the supportpac?

SME [Answer 10] Yes, well there are a number of unique aspects combining significant advantages from the PHP implementation with technology from CICS and z/OS. The PHP engine in the supportpac is precisely the same technology as used in WebSphere sMash. This implementation of PHP is built on top of the IBM Java Virtual Machine. This has a number of benefits. In both CICS and sMash this means that Java can be used as an additional 'system programming' environment and existing Java classes can be exploited in PHP applications. This is how we implemented the API to access existing CICS programs since it was very easy to use the JCICS LINK interface with a PHP friendly front end. This provides the potential to widen the CICS services available to PHP programmers as we learn what is important to them. As scripts execute in the JVM and all JVM workload is eligible for execution on the System z zAAP speciality processors, this means that PHP benefits from the very same price/performance characteristics as CICS Java applications do. The supportpac requires a number of features from the Java 5 SDK, and this is one reason why we can only offer this to customers on the latest release of CICS TS with the PTF applied that enables Java 5, but I think this also a good demonstration of our commitment to Java technology for creating, deploying and running applications in CICS TS using Javabased technologies. As well as the exploitation of zAAP processors, the 'continuous' mode of management of JVMs in CICS has a striking similarity with the execution and management model of the WebSphere sMash runtime. The supportpac also fully exploits the latest features provided by the CICS Web services pipeline that customers are probably already configuring and using to support SOAP, so we are benefiting from the investment IBM has made in that feature - so now we have SOAP Web services, Atom services and PHP services all being accessed by our common pipeline feeding both HTTP and WebSphere MQ messages into modernised CICS applications.

HOST: [Question 11] I can see that this is a very exciting addition to the capabilities of CICS. How can customers most easily get up to speed with it?

SME [Answer 11] The supportpac is available as a download linked from the CICS homepage. With that download comes the executable for the PHP engine itself and a comprehensive guide to installing and configuring the engine in CICS TS version 3.2. There are samples to help you validate your set up included in the package. There's lots of PHP material on the web if you're not already familiar with the script syntax so it should be very easy to start implementing your own scripts. We have a fantastic video available on CICS TV and on Youtube that you will hear about at the end of this podcast, where Robin Fernandes demonstrates developing scripts for CICS using CA1S in conjunction with the Eclipse PHP Development Tools framework - otherwise known as the Eclipse PDT. If in doubt, then simply searching the web for "CICS CA1S" will find lots of information. We often offer technology in the form of a supportpac in order to gain an insight into the customer demand for that technology - this is how SOAP Web services started and look how successful that has become - so we'd very much like feedback on both the general business value of scripting for CICS applications and this specific implementation. Customer can have a dialogue with IBM on this technology via a forum we have set up for this purpose - again this can be easily found via a web search for CICS and CA1S.

HOST: Ian, Thank you. That was really interesting.

SME: Yes, well thanks very much for giving me the opportunity to talk about this exciting new technology.

HOST: Well, that wraps up this podcast discussion. To find out more about the SupportPac for the CICS PHP Support and the demo I mentioned earlier, please go to the description for this podcast at:
<http://www.ibm.com/software/os/systemz/podcasts/websphereonz/>

Join us next time as we talk about another important mainframe topic. For now, this is Sherrie Abshire saying "Thanks for listening".