Hyperledger Update

IBM developerWorks Open Tech Talk

Dec 14th 2016

https://developer.ibm.com/open/video
s/hyperledger-update-tech-talk/

>> JEFFREY BOREK: Hello, good day, welcome to the final developerWorks Open Tech Talk of 2016. I'm Jeff Borek, worldwide program director and open technology for IBM, and I'm pleased to be here along with my colleague Arnaud Le Hors to talk about the evolution of the open Blockchain project. It is something that started in IBM research a couple of years ago, and came out into the light of day on developerWorks Open as a project, and so we are going to talk about the evolution of open Blockchain to HyperLedger. But just before I hand it over to Arnaud I'm going to make a couple of comments about the developerWorks Open program.

Open is something that is in IBM's DNA. We have been doing open technologies for over 15 years now, and of course many people associate IBM with billion dollar
bet that IBM made behind Linux some time back. But IBM has had a long standing history of participating in open technologies and communities and doing so in a really innovative way. Folks may be aware of the huge growth in open technologies is just breathtaking. We touched on Linux but there is also the open stack project, that's got over 54,000 community members. The huge evolution of the node community, which IBM has been helping to foster along over the last two years, and most recently, a HyperLedger, another great example that Arnaud is going to go into detail.

The bottom line is that again IBM has played a significant role in helping to foster these initiatives, not just under the banner of open source software, but also with an eye on open governance and diversity in these communities, to ensure that they are healthy.

Our IBM itself has been exploding in its development of open source software. It is kind of one of a, IBM's best kept secrets but if you go out on just GitHub, IBM.GitHub.I/O you will see there is over 1,000 repos up there now with different projects from IBM out in open source.

With that great volume of content, it can be
challenging for someone who might be relatively new to the space to kind of wrap their head around what IBM is up to.

About a year and a half ago now, we launched a developerWorks Open and we announced it at Oz con and at the time we had about a little over two dozen projects up. Quickly we grew that to over 50 in the first couple of months. By the middle of this past year, we celebrated the first anniversary of the initiative with over a hundred projects in just a short 12 month span.

As you can see, developerWorks Open is really all about highlighting and nurturing IBM's open source projects, not just tossing them out over the fence and hoping someone notices, but trying to help shine a light on innovative and creative new projects from IBM that have high potential.

These projects that end up on developerWorks Open get some extra care and attention in the form of videos, blogs, and developer stories, profiles of the actual code writers that are working on these projects, and after a year, in addition to having over a hundred projects up on the site, we have had a number of projects graduate. And as you can see not just the open Blockchain project but also things like Node-RED as
well as the system ML have all gone to either efforts in the Apache incubator program, or in the Linux Foundation as part of the working group.

You are going to continue to see IBM innovate in open sourced and lead in open technologies and partnerships and part of what we will do now is we will highlight the fact that, it was back in April that after a month up on developerWorks Open, the project gravitated to the Linux Foundation to become one of the fastest growing communities and collaborations on the Linux Foundation.

So with that, I'm going to pause and stop sharing and hand it over to my colleague, Arnaud, to carry the rest of the presentation.

>> ARNAUD LE HORS: Thank you, Jeff. Welcome, everyone. Very pleased to have a chance to talk to you today. So as Jeff said I'm going to talk about the HyperLedger project which started just a year ago. It has, I'll try to give you a bit of a overview for the project is about, and get into some of the details of it, the organization, how the work is organized, but also some of the technology that is being developed.

Let's get started. In my agenda for today, I will try to cover quite a bit of ground, because first I'll
give a short introduction to Blockchain, obviously you are not going to come out of this as being a Blockchain expert if you are not to start with. But my experience is that even though it's a hot topic, there is still plenty of people out there for whom it is new, and so I want to give you introduction, so at least you have some general understanding.

Then I'll talk about the HyperLedger project itself, that is the project that I was just talking about that we had launched over a year ago. Then I'll get into the details of how far we have reached, how far we have gone so far in the case of the HyperLedger fabric which is one of the projects that is being developed within the HyperLedger project organization. And because I know we have a lot of people among the audience here today that are quite tech savvy and probably want to have kind of technical information and how to get started, I will try to get you into some of the basics about how to get started with the HyperLedger fabric, so that out of this presentation, you should be able to go and start playing with it and hopefully go from there.

So, Blockchain, what is it? Essentially, Blockchain is a new style of decentralized database.
A lot of people know, have heard of Blockchain because of Bitcoin. Bitcoin is one specific type of Blockchain. It is one specific Blockchain network, which is public, which is anonymous. When we talk about Blockchain, it's a marginal concept, it's if you will the underlying of Bitcoin. There is no doubt that Bitcoin was the pioneer in this space. But essentially, what people have figured is that there was actually a technology that was much more broadly applicable than just doing currency exchange or currency payments with digital currency.

But the concept of shared ledger that was pictographically protected had many applications, independent of the type of information you store on that ledger. The basic structure is a Blockchain, so as its name indicates, it is a back linked chain, essentially a set of transactions that are submitted to the ledger are packaged into blocks of information that block points to the previous submission, to the previous block, and it is hash encoded.

If you modify any of the blocks, it will actually affect the hash code, and you will be able to see that the data has been tampered with. The database is, there is no way to modify the information that is
already in the ledger. The way you modify it, if you need to, is by essentially submitting a new transaction. If you want to undo something, a previous transaction, the way to do it is you submit a new transaction, the results of which would be to undo the effect of the previous transaction you tried to undo.

The database is basically built on top of a peer to peer network. So that means there is a need for mechanism to synchronize all the different logs. If a node in the network is going to have a copy of the database and that database is synchronized through system called consensus, there are different types of algorithm being used. It is a whole field of research in and of itself but essentially the idea is because you are using peer to peer network to transmit the information across the network, there is no guarantee that the transactions that are being submitted across the network they will all be received in the same order.

The goal of the consensus, all of the Blockchain is to order the transaction and make sure that everybody has the same view of the world. That is my short introduction. Let's move on. What is the problem with traditional databases or if you will what is Blockchain good for?
If you look at typical business scenarios, involve different entities, playing different roles, executing different tasks at different point in time in the execution of the contract. In doing so, they exchange a lot of information. Typically this information is stored in different databases. Some of them can be connected. Some are not. Inherently nobody has any reason to trust each other. They all save their own data in their own format, in their own databases.

So we end up with this complex network of islands of disconnected databases, which implies that you have to constantly try to reconcile those different databases. Another way to think about it is, there isn't one view of the world, of the state of the world. There is multiple views, and it's this reconciliation is both time-consuming and expensive.

Here comes Blockchain. Blockchain essentially is here to provide the ability for competing, multiple competing parties to securely interact with the same universal source of truth. That is what it is about. The idea is what if we all had the same database, we can all look at the same way, and guarantee that people cannot tamper with the information that is in it.

That is what Blockchain is all about. At that
level, you will note that it's a very generic concept. It does not again have to be tied to crypto currency like it is the case in Bitcoin. You can imagine sharing all sorts of facets. It can be a purchase order. Of course it can be also payment. But it can be the certificate indicating that a product was shipped or even before that, manufactured, and received.

So every step of the way of a business, typical business transaction that involves different parties can be captured in a common ledger that is shared by all the different parties.

There is an aspect to this which is important is this notion of smart contract or we call it chain code. Chain code is essentially the programming version of the business contract that parties would typically have in a business network.

Associated with the database we will be able to encode the different terms and conditions of the contract and ensure that the transactions as they are submitted to the network are in the right order.

There is a lot of block chains out there. But what we are especially interested in the case of the enterprise, it wouldn't surprise you IBM was especially interested in the application and for the enterprise,
there is focus on privacy and confidentiality. There is a focus on performance, scaleability. Performance aspect is something that is very key, for those who are familiar with Bitcoin, Bitcoin you have to know that there is like two main categories of Blockchain networks. There are those that are known to be permissioned and those that are unpermissioned or anonymous. Unpermissioned networks, anybody can participate in the network. This is the case of Bitcoin. If you want, you can join the Bitcoin network right now. You can download the client, run it on your machine, but that means by definition nobody can trust you. So they are not going to. And the system is geared to protect itself against rogue agents, people who are trying to screw with the system essentially. The way it is done is small but involves a system that is expensive called proof of work, which is also related to data mining that is mostly known, even though data mining aspect is just a side effect of this process. But it is very costly, very expensive and it has actually by definition time-consuming. In fact Bitcoin is set so that it takes ten minutes to validate transactions. There is no way around this. In fact, as hardware improves, they keep increasing the
cryptographic challenge that is being used as the way of ensuring proof of work, so that it always remains around ten minutes.

Obviously, in the case of the enterprise, this doesn't make much sense, because first enterprise want to, need to have better performance than ten minutes, but also in practice, in the business network, they don't deal with that anonymous entities. They deal with business partners who are known.

In the case of Blockchain for the enterprise, we are more interested in permissioned network, where we can identify the parties and therefore there is a element of trust associated with the people participating in the network. We can control access to the network. And therefore, we can skip some of the pertinent associated with those public anonymous networks. I've been talking about networks.

I already made the difference, tried to make you understand the difference between Bitcoin and Blockchain in general. I also want to indicate there is a difference between Blockchain and Blockchain network. When I say Blockchain, I really refer to the Blockchain technology. That is just a specific new type of technology.
Blockchain network is basically an instantiation of a Blockchain framework. So typically, it can be several networks and there will be specific to one type of application and maybe a specific to particular business network.

I mean in the case of the enterprise, you might say, well, we could deal with proprietary applications or solution. We are still interested in having a system that is completely open. There are several reasons for this, one of which is the system involves high security aspects and security is much better when it's open, open source has shown that repeatedly. But also, it involves a network of people, of parties and it's much easier to get adoption among a set of different players if the system is open, than if trying to, you know, have everybody agree on buying one specific proprietary solution.

So let's move on. As I was just talking about, there is this notion of Blockchain networks which are, there is different instantiations of block chains. Think again. Blockchains of the database, we are not going to have one big database. There is probably going to be lots of different databases. They will all have different characteristics as well. Some will
probably be private. Some will be semiprivate or completely public. They will involve different types of players. So the big vision in the end is that we expect just like there are probably millions of databases out there today, there will be millions of chains out there eventually. We really are at the very beginning of this trend.

Just like the Web, you have a mix of public website, there are websites that are behind the firewall, we will have the similar mix of things. That is the vision we start from, and it's very important to keep that in mind. You will see that there are many people, I mean like Bitcoin community for instance, seems to believe that you can do everything with Bitcoin network and you don't need anything else. There is plenty of new system coming up all the time, you know, that clearly shows that they are not going to be able to stop this.

This brings me to the point about the fact that there are actually a lot of existing Blockchain frameworks already, some of which are just networks like the Bitcoin, there is one specific network that you join if you want to interact with. There are others that are just like their own networks but they are also
the software is often open source. You can download it and install your own network with it.

However, when we look into this again with this, the mind-set of trying to address the needs of the enterprise, what we found is that a lot of those block chains had limitations that didn't make them applicable or well suited to the needs of the enterprise.

Again I'm talking about performances for instance but also problem of privacies, problem of confidentiality, and in Bitcoin the network shares the same ledger which means that every transaction that is submitted to the network is transmitted to everybody. Everybody can see it. That may not be suitable for everything.

There are ways to work around this by adding cryptography on top of this. You encrypt the data that you put on the system. But there is still issues with the fact that people may still track who is submitting what, for instance, and so all of this means that we are not necessarily applicable or suitable for enterprise development.

This led us and a bunch of other people in the industry to create the HyperLedger project. The HyperLedger project is a open source initiative which
has a very broad goal of advancing the state-of-the-art in terms of Blockchain technology across the board for different industries, whether finance, healthcare, insurance, whatnot.

As I was saying earlier, the project really got started in December last year. It was first announced and we had our first meeting in the beginning of January. The project has been going on strong since then.

We are basically, it is a anniversary date and it's a good time to look at what happened and where we are now and where we are going.

The project you have to know is hosted by the Linux Foundation, so it's a, it's known under the term of collaborative project. The Linux Foundation is well-known for its role in advocating for Linux of course, but they have evolved into what I call the meta consortium. It is a consortium of consortium. You are probably aware if you want to create a consortium there is a fairly high overhead to create consortium. You need to set up bylaws, you need to define membership dues, deal with this kind of problems, you need to set up an infrastructure, etcetera.

The Linux Foundation found a niche by saying we
can do all that for you, and so essentially they have
like a class of consortium and they can do new
consortium for you any time you want, and they can spawn
a new consortium at a fairly low cost, in exchange of
some money of course. It is important to know that the
project is actually completely open, even though there
are members that are paying, that are supporting the
project financially, like IBM and many others.

You don't have to pay to participate. The project
is totally open. Everybody is welcome. All the
meetings are public. So it's a very open organization.

If you look who supports the HyperLedger, I'm
happy to report that we have seen an explosion in
membership, even though as I said it's, you could
participate without even paying, a lot of people feel
compelled to actually join, become an official member,
we have a multi tier membership system, with different
price associated with it. But really what members get
is primarily, there is different levels of structure
in the organization, I'm not going to get into all the
details here, but essentially, people who actually
choose to pay, they do that because they believe in the
importance of the project, but not because they need
to have access to anything, or they get a little bit
of visibility associated with the project. Of course there is some marketing opportunities.

And you can get a bit more say in the general direction of how the funds are being used in the organization, but it doesn't go much further than that. As I was saying, membership numbers have exploded. In fact, the Linux Foundation has never seen something so successful growing so rapidly, they have seen 200 percent growth. We have at this point reached over a hundred members. Everybody is very happy with this. If you look at this slide, you can see a whole bunch of names that, with a very broad range of companies including technology vendors but also financial institutions and many others. There are people like, the name escape me, of course, the Anderson industry is looking into this and they have a interest in the managing the tracking of items which is very important in aviation.

The HyperLedger project, I also have to clarify because a lot of people think, okay, so the HyperLedger, they are developing a Blockchain framework. It is a bit more complicated than this in fact, because the HyperLedger project to be honest some of us thought this was going to be just that initially. But it is actually
morphed into a more complex structure, which means that today the HyperLedger project is more of a umbrella organization with many different projects within this umbrella.

So there is common grounds that we all share among the different projects, set of rules about how you participate. They all are centered around a specific open source project, so there is open source element is always critical to any of those projects. But there is in fact and I'll go into details later on the different projects involved, but there are different frameworks being developed, which is why later when I talk about the HyperLedger fabric, I have to specify I'm talking about fabric because if you say I want to install the HyperLedger on my machine, that doesn't mean anything.

I want to make that clear so that everybody understands what we are talking about. There are as I said the characteristics that will be true across the different projects, so for instance, all the projects use the same software license, and we try to promote some homogeneity in the way the work takes place, the different tools being used. So there is definitely a lot of sharing between the different projects.
Let's move on. One other key aspect is a recognition of the fact that, as I said earlier, when we are taking such a broad agenda of trying to advance the state-of-the-art of Blockchain technology, this in general and not for a specific application, it means that you go through the use case and requirements, they are very broad and varied, and you cannot expect that you will be able to have one solution that fits all the problem.

So, that means we have from the get-go agreed that we needed a system where we would have a lot of variability built in to the systems where people would be able to configure the different Blockchains frameworks, so that it is modular and people can pick and choose, and you do some configuration work, be able to suit their specific needs.

So there are some key concepts that I want to go over with, I already introduced a little bit the concept of Blockchain in general. Here I'm going to go a little more specific. There are four main concepts that you need to be aware of when you start thinking about Blockchain in general.

There is this notion of shared ledger, that is what I talked about initially. This is the database itself
if you will, where the data is stored, and this notion of Blockchain. In practice, I'll go more in detail on this, there is a specific structure associated with this and there are variations based on different approaches being taken that are important to be aware of.

But essentially, when you talk about Blockchain, there is always this notion of shared ledger. There is also as I said earlier, this notion of permissions. In some cases there is no permission like in Bitcoin, it's open anonymous. But in other cases you want to be able to deal with permissions. In fact, we have learned and I'll go through this more later, but there are different levels of permission that people want, even more than we expected initially. So we have to go further in that respect.

As I said earlier also touch on this notion of smart contract, which is basically the programmatic version of the business contract that you are executing over the Blockchain network. And this notion of consensus that I also touched about, on earlier, which is the way to reconcile all the different transactions being submitted concurrently to assist them which is inherently asynchronous and therefore needs some kind
of orchestration to organize so that we all in the end, all the participants in the network have the same vision of the truth.

So, there are many benefits to the system obviously, compared to the old model that I described earlier, with the different databases, that are disconnected. You can expect benefits in terms of time, transactions can take a lot less time than in the traditional systems, they require resynchronization, etcetera. But there is also a lot of savings done in just reducing the overhead and the intermediaries in traditional systems, because here everybody can directly participate and see the same data.

Then there is this notion that is built into Blockchain which is tampering resistance and that is again another aspect that is important, the fact that people cannot modify the information that has been submitted to the ledger. I won't go into the detail but just so you know, there is some level of, I shouldn't say disagreement, is it maybe a bit strong but there are different views as to how resistant a specific solution to Blockchain is. You may hear of people talking about tamper proof, tamper evident, tamper resist ability. I'm not going to get into the details
but this is not a black and white thing obviously -- resistant. There is a whole range between something that is completely weak and have no security to something that is super strong.

Without going further into this for now, let me get into a little bit of what is going on in the HyperLedger project in real, in practice. Work is organized in two different ways. There is a set of working groups and a set of different projects that are working on implementations.

The working groups actually are not specific to any particular implementation. We currently have about five different working groups. There is one that is about just requirements. The idea is to gather a bunch of use cases and requirements that are relevant to the project, and that the different projects should be taking into account and trying to address. This is a white paper. This is a group of people volunteered to get together and trying to define at a high level what the HyperLedger project is all about and what this notion of Blockchain technology is really about.

As I said earlier this is still very new, very young technology. So there is plenty of work to do there. There is also a working group working on the
architecture. The idea is to define the reference architecture. There is all this different modules, different concept that I talked about already, and the idea is to define one reference architecture that would put all the different pieces in place, and so that all the different frameworks, well, first we would share a common vocabulary when we talk about Blockchain, and then the different approaches that can be proposed could more easily be compared, because they could be able to, they will be able to document their approach based on the same reference architecture.

There is a working group on identity, so they actually focus on problem in general with identity management. There is actually two facets to this. One is in a permission network you need to deal with identity of the participants, and it's not a easy problem to solve. So there are different approaches possible. There is a set of people who are focused on this problem.

But there is also this notion of, can we use Blockchain to solve identity problems in general. So there is two aspects to this. They are not necessarily exclusive, because you can imagine using a Blockchain solution to solving the problem of identity in the
Blockchain network, in the permission Blockchain network.

There is also another working group working on protocols. It is more like at the level of the network itself. There is obviously peer to peer network as I mentioned earlier. There is a lot of aspects that are related to how the different nodes are connected, what kind of messages are being sent around, and so there is a group looking into these aspects.

Again, those working groups are not implementation specific. Now I'm going to talk about the project that are actually in incubation today. These are actual open source projects. There is code behind it. You can go and download it and run those.

There are actually currently three frameworks already. These are software you can download and run your network with. There is one that is called fabric, which was initially contributed by IBM. There is another one called Sawtooth Lake which is a framework contributed initially by Intel. It has an interesting take, alternative to the proof of work, which aims at not being so CPU intensive.

There is another project, Iroha, which was submitted by, contributed by a small company in Japan
which is actually, has the backing of many major companies in Japan and which has similar fabric but it is written in C++ while fabric is written in go. Sawtooth is written in Python if that matters to you. There is other projects that are around those different frameworks. There are different tools, different SDKs that are being developed. There is also a Blockchain explorer which aims at being independent of the framework, with this notion that you have a monitoring system that allows you to look at the network, see the transaction going through and be able to look at the Blockchain and browse through it.

As I said, the project is being very successful, it has been attracting a lot of people. So there are more contributions coming. There is one that was just announced recently. There is R3 consortium is a consortium of financial institutions, banks, that have been working on designing a Blockchain network that suits their need. So this is known under the name of coda. And they are going to contribute their project to their HyperLedger -- Corda. With that being said let me get into the details of the HyperLedger fabric specifically.

This was contributed by IBM. This is the one I'm
most familiar with and the one I contribute to primarily.

The situation is, since we launched the project and contributed it earlier this year, we actually managed to issue several releases. There is a stable version which is 0.6, and so if anybody wants to start playing with this, please look for version 0.6, that is the latest and most stable version you can already work with.

But it's important to know that we have already learned quite a bit, and we have shifted the focus to 1.0. 0.6 is no further developed and I'll go into the details of the architectural aspects that are behind this so you have a better understanding.

But 1.0 will bring a lot of more good stuff of course. The general release is planned for March 2017. Of course this is a open source projects. So there can be a little bit of, although that is not specific to open source, but maybe more in open source than elsewhere, there could be a bit of delay. But that is the goal we are aiming to have a new version 1.0 in March.

Let me tell you a bit more now about the technical aspect of fabric. First let's start with version 0.6.
Version 0.6 is made of essentially three components. There is on the application side there are SDKs, I'll go more about that further, but there is the peer, that is the main node in the network, and so this is the thing that is going to, the application is mostly going to be connected to. They will transact with the Blockchain through the peer. The peer is your connection to the network, as well as to the actual Blockchain, the ledger which is stored by the peer.

There is also an element called the membership services. It is basically for the aspect that I talked about for fabric is permissioned, and therefore there is this need to deal with identity and access control. The membership services is this element that you go to, to basically enroll users and be able to get credential. You can get certificate for transaction, to submit transactions to the network.

The peer is responsible for all the different elements that constitute the Blockchain that I talked about earlier. You find the consensus, the ledger, there is also this notion of events as things happen, you can actually it will send events to the peers to synchronize with everybody else. But it will also, that is something you can subscribe to as an application
if you want to, to be informed as what is going on with
the ledger.

Then there is the chaincode. Again, chaincode is
what we call the smart contracts. I'll go a bit more
into this later on. But essentially, you have to think
of it, one way to think about it is like, you know, store
procedures. This is the code that is application
specific that turns the ledger which is completely
application agnostic, when it comes out of Fabric, into
something that is application specific, and this is
where you are actually going to enforce the validation
rules specific to your application, so that for
instance, if you say, if your application is about
payment, if you can control that person who actually
submits a payment from one person to another has the
right to do this, that they have the funds that they
want to pay, etcetera. So there is this aspect of, this
is where you enforce the terms of the contract, and that
the contract is executed in the right order.

So, as I said earlier, we actually learned already
a lot based on that first experiment, and we actually
found that even though we added some confidentiality,
people wanted more confidentiality because in reality
in a business network, the execution of business
contracts often involve different parts and sometimes subsets of the same network. People were not satisfied with just being able to have a private network. They wanted to have some transactions only visible to a subset of the network. 0.6 cannot do this. One of the problems is the peer is responsible for so much stuff, it was responsible for the consensus, for all sorts of things.

We also figured that people wanted different types of databases. 0.6 is fairly limited. There is a simple key that you pair a system and people wanted to be able to query the database and we needed to be able to make a bit pluggable. So we added a system of being able to change the database that is underneath, make it more pluggable.

Then there was also a issue with the fact that in 0.6 when you deploy your chaincode it cannot be upgraded. That sounds a bit silly but that is how it is. Basically, you deploy your chain code and you run it, if you want to change it, you have to shut everything down and restart.

We obviously needed to fix this.

Then there is the notion that there were several point, single points of failure and the membership
services for instance was one of them. So we learned that we needed to fix this.

That brings us to the version 1.0 architecture. You can see that there is one main difference already, the idea is that there is this new box on the right here called the orderer. The idea is simply to realize that, well, this consensus business actually is orthogonal to the nature of the data and the nature of the application.

We have actually extracted the consensus part of the network into separate nodes. Then I'm not going to get into all the details here because I don't have the time for this, but essentially we have divided the work of the peer into several roles that could eventually even split further if needed, so that we could have better performance and scaleability.

I want to spend the last time, the remaining time on giving you actual practical information so that if you want to get started with HyperLedger fabric you know where to get started. I focus on version 0.6. 1.0 is still very much in works. I don't recommend you to try and start with 1.0. You will have more pain than is necessary.

When it comes to 0.6 as a application developer,
it is important to know what exactly you are going to focus on. There are two elements you need to worry about, the application side, on the client side if you will, and there is a chaincode development part.

I want you guys to have that in mind. This is where you are going to focus your work, on the one hand the application side, the client, how you are going to interact with the network, and then how you are going to specify or make the data application specific with the chaincode and how you are going to encode the different validation steps that you want to enforce on your network.

Practically speaking, the best way to get started is actually to use what we call the starter kit. This is a Docker, set of Docker images I should say, and it will give you, it includes the Node.js SDK that is for the client side, and it uses go chaincode. They are different languages being used and there is more coming, on the chaincode aspects. But fabric is written in Go so the built in support is for the chaincode to be written in Go as well.

The starter kit comes in two, with two images that you can download from the Docker hub. One will be the peer, and the other will be the member services.
In addition, there is a custom image that you are going to download and you will build locally, because this is going to include the client side that I just talked about with the Node.js and there is a sample application that is part of the starter kit, so you can actually build it and run it right away, right off the bat. Then you can start modifying it from there.

Of course, the chaincode is, there is simple chaincode to get started. It is a fairly complex system, once you deploy all the different aspects. But you can start in what we call DEV mode. This is a very minimal setup where you only have one peer. You can do away with membership services, so they don't have to worry about security. You can make it anonymous and basically just log in and allow anybody who can log in, you can run having on your machine. You don't have to have several networks, I mean several peers of networks. You can only have one.

The consensus aspects, basically doesn't do anything. It just says sure, you submit a transaction, I'll take it.

Then you can move on from there in different stages, so that you don't have to handle all the complexity at once. Once you have a chaincode that
actually works, it compiles, it runs, you can deploy it. You can invoke transactions. Then you can add the network aspect where you have several peers that connect to one another. You can add the security event where you add this notion of a role where the peers have to log on to the membership services to get certificates, so that they can authenticate themselves and the transaction can be authenticated.

There is this question part that is totally usable and makes your life easier as you get started.

I mentioned earlier that the ledger itself can have different actual structure. It is important to know that in fabric what we have is the ledger is basically made of two parts. There is the Blockchain which is actually managed by Fabric. You don't have to worry about it. This is the actual Blockchain that I was talking about when I was introducing the notion of Blockchain with this different blocks of transactions, so different transactions are packaged together into one black and chain with the other ones. The consensus, all of this figure this out, you don't have to worry about it.

There is the world state. The world state in version 0.6 is as I said earlier, a simple key value
pair database. It is very basic. It is not application specific. That means that as a developer, you are going to, we refer this to the world state and essentially what happens is that the application is going to invoke transactions that will be executing some code on the chain code in the peer, that will change, submit changes of states to this database by changing the value or maybe adding entries to this key value pair to the system.

Again it's very basic in practice, in 0.6 we use rution DB and there isn't much query capability. There is a hack where you can structure your data as a kind of tables and rows, but it is very weak and it's just thin sheen on top of the actual basic structure. So don't expect much in terms of querying capabilities at that table. Essentially it's just a ledger, you put the key, value associated with the key and you can retrieve it. That is about it. The system will take care of the aspects of managing the consensus and packaging the transaction, all of that is done for you. From a chain code point of view there are three key functions you will need to worry about.

There is one that is init, when you deploy the chain code, which means that you tell the peers that
load your chaincode, and then it will actually run it's like an initialization step. That is run once at deployment time.

There are two main functions, one is invoke, and when you actually want to invoke some code that will possibly change the chain code, so this is like the kind of read/write function if you will, while the query is guaranteed to be only read only.

That should give you a bit of a framework to get started. If you need help, you have to know that there is a vibrant community of people hooked to slack, there is a set of mailing lists, and there are also questions slashdot we are trying to populate slashdot because there are a lot of questions that keep coming back so we are trying to provide people with basic function, information to get started.

I want to note though that I gave you a basic way to get started, so that you can right away get started and be on good path to a good experience.

But fabric comes in many different flavors. You have Docker images that can be found on BlueMix which is the IBM open Cloud. So if you don't feel like running on your machine, you can just as well log into BlueMix and launch, there are demos available there.
You can quickly run a low Blockchain network in BlueMix and start from there.

From an application integration point of view, I talked about the SDKs, Node.js which is part of the starter kit, there are other SDKs available. There is one in Python, there is also one in Java. The main API is in GR PC because that is based on go but there is a REST API that you can use. There is a common line interpreter that you can use to interact with the system to launch peers. You can do everything from this CLI so that might be something you also want to look into.

I want to highlight the fact that you don't have to start from scratch. Don't be too intimidated. There are several examples already available. They are easy to find. You can get started quickly at a basic chain code hello world type of thing. You can go from there. You just have to modify the code rather than starting from scratch.

I'm almost at the end of my presentation. Before I move on though, I want to also say for the more experts among you, if you are so inclined, you can of course also contribute to the HyperLedger fabric. As I said this is a open source project, so we very much welcome contributions. You just have to know a little bit
about what it would take. Obviously this requires a bit more than what I just described with the Docker images, that you download. The first thing is you will need to basically have a Linux Foundation I.D. This is easy to get. You go and register and you get an I.D. You can use different types of ways to install the framework with the source code on your machine. We use GitHub with vagrant. There is ways to run it natively on Linux and iOS. I don't recommend trying to Windows. Nobody really does that, if you are on Windows, the best way is to use vague rant -- vagrant so you run a virtual machine with Linux on it. For project management we use Jira, we use Gerrit. We use git as the repository obviously. There is a GitHub which is a mirror of the Gerrit, of the git repo. But we use Gerrit which provides code review that is not available in GitHub.

In terms of your focus, I want to say the sky is the limit. There are many different areas. Block chains are pretty complex systems, and so it's a little bit based on your inclination and your interest. You might want to focus more on chaincode or on consensus. It's up to you. Again there is plenty of help available, if you need to. Don't be intimidated. Don't hesitate. Go on Slack. A lot of people like
myself hang out on Slack any time we can. We are happy to answer a question and help people get started.

Another way to get started is actually to come and participate in one of our hack fest or hackathons. These are events that we run fairly frequent basis, every other month or so, we have a hack fest. These are like HyperLedger meetings where the committee gets together, and as I said we have different frameworks being developed in parallel but this is an opportunity for us to have a, we talk about what is going on in different projects and try to learn from each other and exchange ideas. Also hackathon which are fun events to participate in, and can be a good way for people to get started with hands-on experience.

>> JEFFREY BOREK: Arnaud, this is Jeff again. Great presentation. Thank you. We have just a couple of minutes before the top of the hour. We have been inundated with lots of questions. We have over almost approaching 60 now. We have been able to answer approximately half. To those that were not able to get back to live, we will follow up and try and get back to you to get answers to your questions.

But Arnaud, you mentioned finance and aerospace specifically airbus I think and Boeing is also
publicly, has made some noise about interest in this technology. Are there any other industry verticals that come to mind? Healthcare is one that I've also read that there is some early activity going on and that raises the question of the HIPAA compliance. So anything briefly to mention about other verticals?

>> ARNAUD LE HORS: I think right now, there is a lot of hype around Blockchain and quite frankly, I caution also people because I'm old enough to have seen a lot of new technology come up, and everybody, every time, thinks that oh, this is it, this is magic technology that is going to revolutionize the world. And we are going to solve all the problems.

I think there will be a lot of misuse of Blockchain at first. But it's fair to say that it's a reasonable thing for everybody to actually have a close look, get familiar with the Blockchain technology in general and try to make an assessment as to whether they have a problem that is well suited to this.

For instance, governments are looking into it. I talked to colleagues here, I'm in Europe, and people are looking into for keeping track of products being shipped around Europe, where they have custom information to deal with, taxation issues, to figure
out who owes what to whom. And there is plenty, I mean quite frankly there is a lot of applications that are possible. We really need to understand what is most suited.

>> JEFFREY BOREK: Can you spend a minute talking about some of the competitive technology that is competing against this? There were a couple of different people asking about alternative options or competitors. If you have any just top of mind thoughts briefly, that would be great. While you are doing that, Clyde, could you post the URL for the white paper into the chat, so the link is live. Some are pointing out that some of the links on the slides that they are seeing they can't get access to.

>> ARNAUD LE HORS: These are not clickable unfortunately.

>> JEFFREY BOREK: Clyde will get those up while you answer the competitive --

>> ARNAUD LE HORS: Yeah, so as I said earlier there are many different projects going on. I'm glad to say that the HyperLedger has been able to move into the system where we have actually some kind of competitions/cooperation going on at the same time -- HyperLedger. And it's a very friendly
environment, where people actually develop what could be seen as competing solutions in parallel. But like I said we are trying to learn from each other. This is all done in very good faith.

I think there are projects like Bitcoin, I said earlier they are trying to expand beyond currency. There are people, quite frankly, I would say religious about this, they think that anything that is permissioned is a heresy. Part of it is because I talked about the machine services and it's a form of centralization which people say, this defeats the whole point. You are trying to build decentralized system. Some people think that there is no saying into having this kind of private networks where you are trying to trust people and you don't have the system like proof of work that will guarantee that people cannot alter the information, there is no way you can actually rewrite the history.

Quite frankly, the HyperLedger project is trying to be, is very agnostic about all these things and we entertained all the different aspects. So I talk about fabric which is permissioned but so you know, it's not the only proposal on the table. So two click which I mentioned earlier is unpermissioned earlier so it's
more like Blockchain. Sorry, Bitcoin.

The other project like etheram which is well-known and I talked to some of their community members. I talked about new projects that might come to the HyperLedger project. Some people in the etherimcommunity are thinking of bringing some of the components into the HyperLedger.

I think it's a field that is in explosion right now. There is plenty to experiment with. So I think it's very healthy. I wouldn't be worried about it right now.

>> JEFFREY BOREK: For example, somebody asked about the batchly project and Microsoft has a initiative around this technology that is similar but it's really not a community at this point in terms of type of broad working group that is under the Linux Foundation banner of the HyperLedger projects.

>> ARNAUD LE HORS: Right.

>> JEFFREY BOREK: With that, we are beyond our time limit. I'll apologize to our audience where if we didn't get a specific answer to your question, we are going to try and get back to you, as part of the post-game analysis. We thank Arnaud, our presenter today. We thank our audience for the DeveloperWorks
Open Tech Talks. It has been a great 2016. We look forward to getting back together with you, look for the next tech talk coming in early January. It is going to be on the Eclipse OMR update, that is the first call of 2017.

I will share the date, if my screen renders quickly enough. That is going to be January 11 at the same time. So thanks, everyone. Have a great rest of your year.

>> Thanks, everyone. Bye-bye.

>> Thank you, everyone.

(end of session at 11:02:00 a.m. CST)

Services Provided By:
Caption First, Inc.
P.O. Box 3066
Monument, CO 80132
800-825-5234
www.captionfirst.com

***
This text is being provided in a realtime format. Communication Access Realtime Translation (CART) or captioning are provided in order to facilitate communication accessibility and may not be a totally verbatim record of the proceedings.
***