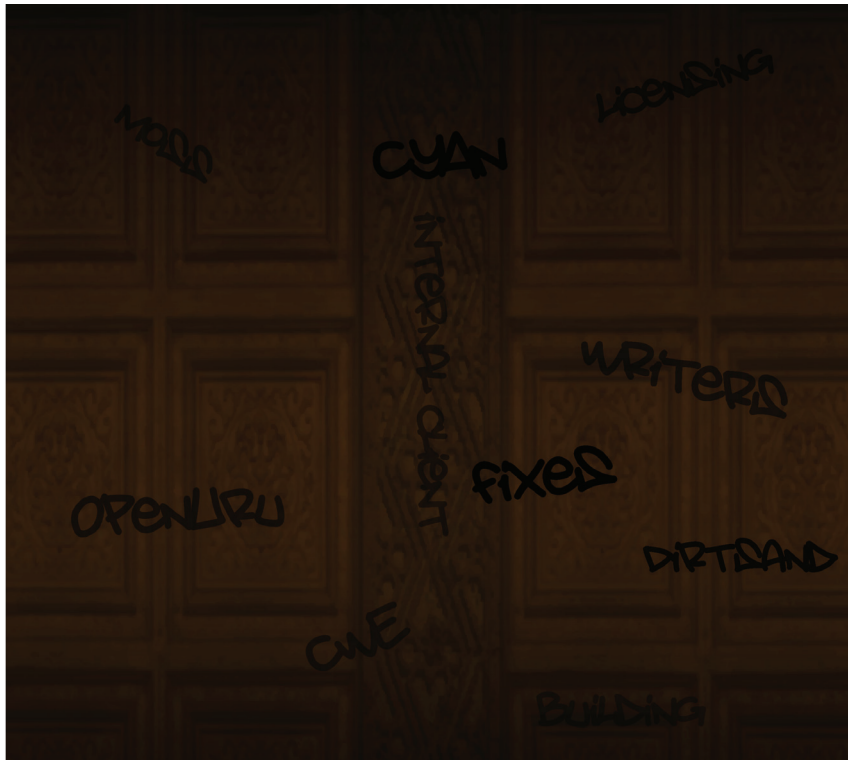


---

# Behind Open source

---



Ever since Cyan Worlds announced their intention to move MOUL to open source there has been eager anticipation from fans for the change. On April 7th Cyan released the Cyanworlds.com engine along with development tools through OpenUru.org. There were many surprised reactions from fans due to the fact that very few people knew the exact date things would be ready. However the announcement from OpenUru.org is just the beginning. Cyan still needs to release the MOULa server code. The completion of the Myst Online Server Software (MOSS), which is a replacement for the MOULa server has been released as an open source project as well.

There are still many people working on improvements and other important additions which will be needed to create a successful open source project. Many in the GoW have been frantically coding ever since the source was released just to bring things up to speed. Several fan made changes have already been incorporated directly into the current MOULa client. It is very likely that more fan made changes will be incorporated into the current MOULa client. Eventually there will be entirely separate shards being run by fans. This issue is dedicated to the open source release and will reflect on the future possibilities for development.

-Brad "AgeExplorer" Christensen

# Glossary

**CWE** = Also known as **Plasma**, it can refer to both the game engine and the client used to connect to the game.

**Bullet** = Open Source, multi platform physics engine. It was used in Plasmaclient and might be used to replace Physx.

**DirectX** = handles multimedia tasks like video and makes the game visible on your screen

**Havok** = Physics engine that was used in Uru:ABM, Uru:PoTS, Uru:CC, Prologue and UU

**MOSS** = Myst Online Server Software, has been developed by a'moaca' and cjkelly1

**OpenGL** = Open source, multi platform replacement for DirectX

**Open Source** = In this particular case, it means that the code for the client/CWE and the MOSS server have been released.

**Physx** = A physics engine from Nvidia implemented before Gametap but after Until Uru etc. A physics engine takes care (among other things) of the kickables like the cones.

**Plasmaclient** = Multi platform(Windows, OSX, Linux) Open Source client, developed by Branan. After open source the project was abandoned.

**Pyprp (2)** = Blender plugin to build ages. Pyprp 2 is the version for the newest Blender(still in Beta) and is still in development.

# Open about Open Source

## The Beginning

On April 6th, Mark “Chogon” DeForest posted an announcement in the Myst Online: URU Live forums. It was a letter from Myst co-creator and Cyan Worlds CEO Rand Miller to all of us in the community. Thanks to the work of Cyan and a handful of dedicated individuals, open source URU Live was now a reality. To say this announcement sent shockwaves throughout the community would be an understatement. This is something most die-hard fans had been waiting for since the closing of the Gametap servers, the subsequent regaining of the rights to Myst Online by Cyan, their first announcement of plans for open source, and the reopening of the cavern in February of last year.

Their “cautious optimism” had been rewarded once more.

How exactly did open source finally come about? Of course, we all have questions, as this had come quite out of the blue for most of us not already in the loop. For this, OpenUru.org was sent questions by AgeExplorer of the Guild of Messengers, with many of the key individuals involved responding.

Work for open source began on or around December of 2008. Despite their great desire to get the material out to their fans as soon as they could, Cyan had to focus first and foremost on paying projects. One such project was “MagiQuest Online”, which brought JWPlatt of Creative Kingdoms into a business relationship with Cyan, and frequent discussions with Chogon.

“I remained in contact with Mark (Chogon) fairly consistently after MagiQuest Online Chapter One was completed”, said JWPlatt, “and the topic of open source would come up occasionally.” He, along with Mac\_Fife from the MOUL forums, kept open source on Cyan’s presumably long list of things to do. After the release of the “Riven” port for iOS on December 15, Chogon felt it was the right time to push to get open source out. And so, about two days later, he sent an e-mail to Rand, Tony Fryman (President of Cyan Worlds), and the OpenUru.org team detailing the latest plan for open source.

## Crutial Component Replacement

One problem was the fact that there was no server code to be included with the rest of the open source tools. While the release of just the client and plugin sources would have been a large step in and of itself, source code for server software was needed to turn that step into a giant leap. And Cyan was still far from ready with any offering of their code. Enter a’moaca’ and cjkelly, developers of MOSS (Myst Online Server Software), which was built inspecting the network traffic between the client and Cyan servers. There was never any access to the original server code for the development of MOSS. In a case of excellent timing, soon after this new push for open source began, they introduced MOSS to the team.

“MOSS was an incredible gift,” JWPlatt said of their work. “The original plan

contacted us about hosting MOSS not long after we started our open source release plans. So now we had a working MOUL server and both a 'moaca' and cjkelly aboard to make the client work. We suddenly had the capacity to offer a complete MOUL shard."

"They have helped me and Cyan numerous times," added Chogon, "including that dodgy network problem with Gametap MOUL service (and of course, Mac[-Fife] was involved as well, heh)." The fact they were able to create such a project was no surprise to the Cyan employee, as he had met them previously during Mysterium, an annual gathering of fans, at Cyan Worlds' offices in Spokane, WA. "I had to pry her and cjkelly away from a wayward printer at Cyan they were going to set right. Not that I didn't think they couldn't fix it, it just didn't seem right that fans on vacation should be working on such stuff. But of course, digging deep into the internals of things is the way of a 'moaca and cjkelly."

For a 'moaca', MOSS work began simply for fun. "MOSS started out as a smaller project both to let me 'keep' the MOUL ages, and to see what the interesting parts really are in a server project." Over time, her pastime grew into a full-fledged multiplayer server project.

As would be expected, it was not easy work. There were many roadblocks in the development process. "Sometimes we had to just write something and see what happened when you put it all together and tried to play the game", a 'moaca' said of the troubles. "We got stuck for months on a bug where avatars would be invisible to others, but in a very asymmetric way." Despite this, the

sometimes tedious work continued and progressed until they had something to present to Cyan, and as a result, created a way to alleviate the pressure on the company to release its own server source code.

## Putting It Together

Another concern raised by the OpenUru team was how exactly to organize the distribution and building process. Surely, a centralized collection of tools would be needed for a bit of consistency in the course of development. Rarified, a later addition to the team, spent the time that Cyan and the others were working through the legal details to create the framework for just such a collection called "the Foundry". Said Rarified, "I had hoped that this shared set of tools and processes would be attractive both to the MOULa community, making it easier for non-technical people to have access to current versions of the game, as well as to Cyan by having a well known place for collaboration and sharing new (and fixed) elements of the open source game."

Despite having the repositories, bug-tracking, and source code browsing and review tools at launch, Rarified mentioned there is still more that can be done with the Foundry. "The infrastructure to build the source code into programs you can run on your computers exists," he said, "but still needs to be configured to monitor the repositories and perform the build tasks." There are many other options being considered for his part of the project, including creating automated tests of new builds and to even make available, through a shared environment, tools most don't readily have access to, such as 3DS Max.

## Much to Be Done

So what happens now that open source has been unleashed upon the public? Everyone from Cyan and OpenUru makes it clear that, despite the release, the real work is far from over. Debatably, the most important issue is properly replacing the code that Cyan needed to take out due to rights issues. “In stripping out the code that Cyan couldn’t redistribute and substituting alternative libraries,” said Mac\_Fife, “some ‘emergent properties’ have arisen, not least several oddities with PhysX. That’s on top of things that might already have been in Cyan’s build of the software.” He goes on to say, “We feel that the first thing that needs to be done is to try to get back to as near as possible a functional match to Cyan’s build, so we’ve got a comparable baseline to work on. Rushing to try to add new features before getting the fundamentals working properly would just create more debug work in the future.”

JWPlatt was in agreement on this as well. “[...] we would prefer to concentrate on restoring functionality before getting to the pesky bugs people love to hate.” In the meantime, Rarified’s aforementioned bug-tracker, dubbed “JIRA”, can gather any and all submitted descriptions of bugs and glitches into one central database for future fixing, testing, and status updates. It would be the place for programmers to check which bugs were still a problem or not, and how widespread said problem was.

As for collaborating with others who have already been working diligently on opening up and modifying various

aspects of the game for years - most notably the Guild of Writers - the OpenUru team says it is willing to start professional relationships with anyone wanting to help. As JWPlatt puts it: “We will pursue or invite working relationships with any group of developers or any individual developer. Any venture like this is going to work better and reap more rewards when everyone shares what they do. That’s open source.” Mac\_Fife insists that OpenUru.org is merely a place that “provides tools that we hope will assist developers, age builders, artists, etc., whatever allegiances they may or may not have.”

One of the key points that Cyan, OpenUru, and indeed most fans, have been stressing is that access to all the materials should be as open as possible. However, some restrictions will need to be required in order to prevent breaking of crucial systems. However, they say the process will be open to other qualified people in due time. “[...] we’re in the process of trying to fast track people into position for that,” said Mac\_Fife. It should be noted that, while at first an original OpenUru.org account was needed to acquire the source files, since then, a guest account has been set up to make the process a bit easier (username: guest, password: guest). “We could not predict interest in a project as large in scope as one by Cyan Worlds,” said JWPlatt, “so it was prudent to prepare for significant demand.” Chogon would only request one particular restriction: that one main branch of development be set aside “to be what is on (or completely compatible with) MOULa.”

The most important thing the team wanted to stress was that, while the tools were released and were available initially on their site, this project belongs to everyone in the community. As such, it will take the effort of everyone and anyone willing to help to keep the project afloat. “I’d like to emphasize that we are all Open Uru,” explained JWPlatt. “OpenUru.org can be anyone who participates and contributes, not just ‘they.’ What we’ve done is a start. It needs your (the collective ‘your’) participation. If past demand is any indication, there will be a lot of people who want to participate and we’ll make every effort to give them that chance.”

## Looking Forward

Someday in the future, Cyan does plan to release the actual MOULa server code, and one might wonder if this will conflict with the current MOSS server code in any way. The release of their code could only help MOSS, says a’moaca’. “Certainly the server code would reveal new things about how MOSS ought to work. For example, I’d love to know what algorithm is used to filter kickable messages (e.g. cones). [...] MOSS does it, but not as well as Cyan’s server. So MOSS could definitely learn a thing or two.” cjkelly added, “When Cyan server sources are available, it should be a simple matter to adapt their stored procedures for MOSS, if it is determined that such a thing is desired.”

Perhaps the most important thing in the casual player’s mind is new content. The tools provided will make it easier for such creations to be tested on the more current MOUL client, as before the release, it was regulated to slightly

more outdated “Path of the Shell” and “Complete Chronicles” versions of the game. Testing new Ages in-game will also be easier. a’moaca’ explains, “[...] to change an age’s SDL file in Alcugs [server software], you had to shut the server down and I hated that, so with MOSS, you don’t have to shut down the server. [...] The nice thing about the open source is if we need either the client or server to change to ease age creation, in ways not already present...we can do it!”

Open source MOUL is the culmination of countless hours of work on the part of Cyan Worlds, Inc. and its dedicated employees that have created some of our favorite video games, and have also supported the Myst/URU community for so many years, above and far beyond what most video game companies would allow these days. Also to be thanked are not only the individuals in the OpenUru.org team, but all those who have taken up the cause of helping one of our most cherished games continue to thrive and evolve. With the advent of open source, the possibilities will be near endless due in no small part to all their tireless efforts.

“The ending has not yet been written”... and so it is the same for Myst Online: URU Live. \*\*\*

- Brad “AgeExplorer” Christensen & Patrick “Doobes” Dulebohn

Special thanks goes to, in no particular order: Mark DeForest (Chogon) of Cyan Worlds, JWPlatt, Mac\_Fife, Rarified, a’moaca’, and cjkelly for their participation in this interview.

This article is a summarize of the interview which can be read [here](#)

# A dedicated few

Long before open source was even considered by Cyan Worlds, fans have worked to unlock the secrets and make URU customizable. Playing through all the “official” Ages, while fun, can also get repetitive. After URU Live’s cancellation by Ubisoft and the release of “Path of the Shell”, the game was in danger of becoming stagnant with no new content. Talented programmers, artists, and builders were anxious for the chance to add their own flair into the game and see how their version of The Art would stack up to Cyan’s (or the D’ni’s, if you’re in-character) work. Many groups sprang up to do exactly that, reworking aspects of the game to allow for fan expansion.

One group of note that has done considerably well with such re-engineering of URU and have done amazing things to get new content into, at the very least, the offline version of the game, is the Guild of Writers. Named after the D’ni establishment that, in Myst canon, wrote the fantastic Ages we have seen primarily in Myst V and URU/MOUL, their incredible work, such as PyPRP, has helped to keep the story going, so to speak...and the community interested.

With the recent release of open source, it was only natural that many would turn to the Guild to see what their opinion was and what they now planned to do with the code. AgeExplorer interviewed Branán, a Guild councilor for the GoW and creator of PlasmaClient, which was an open-ended, fan-made alternative to Cyan’s official Myst Online: URU Live again client. PlasmaClient is not

supported at this time.

Perhaps the first question on the minds of fans is: what did Branán think of how OpenUru.org did with their release of the source code? “My big issue with the way Open URU handled things was not having planned for contributors from day one,” he said. “The most important part of running an open source project is being able to attract developers and get contributions. So if you don’t have a plan for developers so you can say ‘here is how you get your code in,’ then you are not going to get code from developers.” In his eyes, the GoW would have had such a plan in place from the beginning, even if it was only a temporary, but modifiable, outline.

To the best of his knowledge, the Guild was not asked if they wanted to bring people on board for OpenUru’s team. “If they did I wasn’t informed.” Branán did not become a Guild councilor until just after the release. That, plus his busy schedule, did little to help him keep up with the latest news, although he remained in the loop about most Guild business.

And as for Cyan, their decision to use OpenUru.org was, reportedly, due to the business ties they already had with a few members of the site, particularly Creative Kingdoms’ JWPlatt, but some expected that they would have also utilized the Guild, as its members had been studying the game’s inner workings for years. “[...] that is kind of what several people in the GoW suspected,” Branán said. “I agree from a business

perspective. That does make the most amount of [business] sense but I'm not sure it was the best decision from an open source project perspective." As Branán mentions, OpenUru did not approach him to utilize anything from his work, but he is hoping that things will be sorted out "so we can be sharing code back and forth at some point."

Despite all that, the Guild is ready to get down to business on its own projects and on its own terms. Plans are in the works for an official Guild of Writers shard, a client to compliment it, updated 3DS Max plugin binaries that will work with more current version of the program, and of course, their highly anticipated release of PyPRP 2. For those unfamiliar, PyPRP is the open source plugin that allows Ages created in Blender, a free 3D modeling program, to be converted for use in the offline version of URU. While arguably not as "perfect" as Cyan's conversion techniques, it has made great strides over the years to bring fan Age creation and functionality as close as possible to such standards. "[...] we have been hard at work basically since we got the source, getting the code to run with updated tools and technology. This is going to be our big release."

Fan Ages will no doubt be a major contributor to the success of open source and MOULa as a whole. Which ones will eventually be included on the official Cyan server is always up for debate. "I think the good ones will definitely be accepted by fans," Branán said. "I think there are a lot of fan ages done by people who are learners and obviously they don't have quite the professional quality and polish. Many of them are still fun to play, but, you know, issues like

colliders not being quite right and lighting being off is fairly common with fan Ages. [...] I think Cyan will very likely take a conservative approach to fan Ages because they're still trying to market a product and they need to keep just the best of the best in there. So I think there will be plenty of opportunity for people interested in the whole selection of fan Ages to run things." He notes that, with new shards coming into place in the future, there will be no shortage of assortments of different content.

Some have said that fan Ages, even the most polished ones, don't have that "otherworldly" feel to them, like Cyan's Ages of Teledahn or Kadish Tolesa. Branán has noticed this as well. "There has been a few Ages that have been pretty out there," he said. "I think most of the better quality fan Ages have been things that either set up to either to be locations in D'ni or garden Ages. The garden Ages, by their nature, tend to be a little more tranquil and Earthy than otherworldly. I think there is potential, there is a lot of imagination in the Guild of Writers."

PlasmaClient was one of Branán's notable creations, and the talk of many of the All-Guilds Meetings held in Kirel and on various forums. When open source was released, many immediately wondered what would become of it. "I think I was always pretty clear with PlasmaClient that it was a project to keep me occupied and entertained until such time as Cyan got the source out. Now I have basically turned all my energies on Cyan's code." With that, PlasmaClient work has ceased, but Branán hopes that some of the features he implemented in his client version work their way into an "artist's build"

utilizing Cyan's code, such as physics draw to see colliders.

One of the biggest goals of the Guild is to get working native builds of the game for both Mac and Linux users, who have, up until now, been forced to either wait or find elaborate ways of getting the current version to work on their systems, with mixed results.

As Branan explained, this conversion will not be an easy task, but the Guild is committed to it, and pieces of his PlasmaClient may be able to help. "As we move towards porting CWE to Mac and Linux (and that is one of our goals because it's clear there is a good segment of the population that uses those systems and it will be nice to have a native build for them), we're going to have to replace PhysX and DirectX, and PlasmaClient is already based on Bullet and OpenGL. Although the core game logic isn't great, the physics and rendering in PlasmaClient are something I can use as a base when working on porting Cyan's code."

During the Gametap era of MOUL, Cyan had to replace the Havok physics engine due to their license for it having expired and a renewal being prohibitively expensive. PhysX was the alternative they used, and the effects of the change have varied from good to frustrating, to say the least. "PhysX is great if you want to create a world where you stack up a bunch of boxes and throw a baseball at them," Branan explained. "PhysX will do that perfectly. PhysX has issues when you want the character to be able to interact with things." Cyan's implementation of the engine is partly to blame, according to him, but that may be due to the fact that Gametap

didn't give them enough time before the launch of MOUL to properly implement, and adding character interaction is tricky at best, to say nothing of any of the other physics in the game. Another reason, he says, was the company probably didn't want a large increase in system requirements, "so they decided on the easy, less performing but also less system intensive solution."

Similar to the OpenUru.org's position on the game, Branan is more worried about getting fully working versions of the game out, particularly to Mac and Linux users, before implementing new features. Once that is complete, however, "I'd like to see updated graphics so that there can be more than just a couple of real time lights at a time, better cube map rendering, you know, just a few basic graphical improvements for bringing the engine into the 21st century. When you look at certain aspects, especially the way the avatar is lit in the game, it is starting to show its age." And while he agrees that most of the static environments in the game still hold up visually to this day, there are still many improvements that can be made.

Because of this, Branan believes that, while it's wonderful that URU can still run on rather old graphics cards, it's time to upgrade the game as some artists will undoubtedly add features that older cards may not support. "There is no way around that," he says, "but I know that some people in the URU community have very old computers and they might not like me saying this, but there's a point where the developers and the artists just can't target really old hardware. It's just a matter of practicality." He would like to be able to find

some way to implement “graceful degradation” for older graphics cards to work, but in the end, the GoW primarily wants to add more options for the artists.

URU has gone through many changes over the years, as has the staff at Cyan, so as such, the state of the code released is of interest. When asked if he has seen anything unusual in the code he’s looked at, Branán had to laugh. “Oh lots of things. The code has been built by many people over the better part of 15 years. It has got some eccentricities, let’s say. There is definitely some really good code and there is some really bad code.” Replacing said “bad code” has been another priority for the Guild, with a vast majority of the networking code in particular being looked over by Adam “Hoikas” Johnson, also of the GoW. His fixes, primarily solutions to a lot of lag issues having to do with communication between client and server (think of that 2 second delay when sending an invite through the KI, the delay when linking to Ages, and other minor yet annoying pauses like those), will end up in the main branch of their work very soon.

Branán, along with others in the Guild of Writers such as Hoikas, Zrax and Paradox (and many others not mentioned but equally important, of course), have all done their part to keep the spirit of URU alive despite many setbacks over the years. While he is not exactly happy with the current state of how open source is being handled, Branán is more than willing to work with OpenUru.org when they get “a plan in place to integrate code and also having some public declaration of what they want to do. Our focus at the GoW has been and will continue to be on what the artists want.” He makes clear that he respects both

them and Cyan Worlds, despite some past bad blood between them and the Guild, and would like to keep working with them. Expect exciting new projects from Branán and all the members of the Guild of Writers in the coming months. They never disappoint. \*\*\*

- Brad “AgeExplorer” Christensen & Patrick “Doobes” Dulebohn

(Big thanks to Branán for taking the time to do this interview!)

This article is a summarize of the original discussion, which can be read [here](#)