

The Gopher Times

Opus 4 - Gopher news and more - Apr. 2022

Molasses Gopher/Gemini Client

gopherml

Jonathan Simpson is announcing a new Gopher client: Molasses.

>> A new gopher client, Molasses, is now available for general use. It is a multi-platform graphical client that runs on Windows, Mac OS, and Linux.

Leveraging functional programming with Racket, the binaries come battery included, bundling the racket

>> Feedback is welcome and appreciated.

runtime code, famous for building-up robust graphical user interfaces straight from the core language libraries.

Inline images, multiple tabs, keyboard navigation, Gopher and Gemini support, opening external http:// links on an external browser, Molasses has everything one might expect to browse the little Internet.

sfeed 1.4 released

Hiltjo

I want to thank all people who gave feedback.

sfeed is a tool to convert RSS or Atom feeds from XML to a TAB-separated file.

It can be found at: [1]

sfeed has the following notable changes compared to 1.2:

Fixes

- Fix a compiler warning with some curses implementations, like NetBSD curses.
- sfeed_curses: add keybinds for the home key and the default home and end key for urxvt.
- sfeed_curses: fix a redraw when reloading a file with a feed file read from stdin and using an URL file and changing this URL file externally.

- sfeed_curses: cast character for SFEED_AUTO CMD to unsigned char to allow character sequences outside the ASCII range.

Documentation

- README: add an example script to count new and unread items. This can be useful for some status-bar indicator (asked about by e-mail).
- Small code-style, comments and documentation improvements and fixes.

Testsuite improvements

The testsuite repo has had improvements to test the most important code paths of sfeed_curses in an automated way (currently 95% automated cover-

[1]
[git://git.codemadness.org/sfeed](https://git.codemadness.org/sfeed)
[gopher://codemadness.org/1/git/sfeed](https://codemadness.org/1/git/sfeed)
<https://codemadness.org/releases/sfeed/>
[gopher://codemadness.org/1/releases/sfeed/](https://codemadness.org/1/releases/sfeed/)

[2]
https://git.codemadness.org/sfeed_tests/
[gopher://codemadness.org/1/git/sfeed_tests/](https://codemadness.org/1/git/sfeed_tests/)

age). The sfeed.c and xml.c parser coverage has also near 100% coverage.

The goal is to find bugs and avoid regressions.

The input/sfeed/realworld/ directory contains files with various feeds from popular systems to more obscure ones. These may be useful to test other RSS/Atom programs aswell.

These tests can be found here: [2]

Thanks, Hiltjo

BBC Reviving the Plain Old Radio

tgtimes

BBC, one of the earliest if not the first radio broadcasting ever, comes back to using a WWII era technology, to overcome limitation Russia imposes over Ukraine.

In between a rain of missiles and a short moment of temporary peace, fetching information on what is happening around is a relief, maybe even a requirement for survival.

Internet infrastructure of Ukraine are being impacted, and the backbone getting shackled by all kind of limita-

tions, provoked the BBC news bulletin to be unreachable.

A more primitive way to broadcast critical headlines than Internet: **short-wave radio**, which can live off a simple emitter for covering a large region.

>> It has launched two new short-wave frequencies in the region for four hours of World Service English news a day. These frequencies can be received clearly in Kyiv and parts of Russia. [1]

Shortly after, possessing a shortwave radio device at home became forbidden, proving that in spite of being a low-technology solution, it was efficient enough to disturb the control of the press by the government.

This showcases how quickly-deployed and resilient simple technologies can be in comparison to fragile, high-tech interdependent ecosystems.

Radio is also trivially interfaced with high-tech: Any person with an analog emitter may start broadcasting a radio signal, reading a news digest out loud.

[1]
<https://www.bbc.co.uk/mediacentre/2022/millions-of-russians-turn-to-bbc-news>

[2]
<https://hackaday.com/2022/03/17/owning-a-shortwave-radio>

Given instructions, a receiver is also very easy to build with scavenged parts. An antenna is simply a wire producing an input signal, that after demodulation, becomes a sound signal to be fed to a speaker.

It also shows benefits of putting all the technically difficult parts onto the side of the content producer. It helps with adoption of a new technology: Making the client device/software trivial and safe to build, setup and use. [2]

New Bitreich Project: rfcommd

20h

There is a new project on bitreich: rfcommd. Rfcommd is a daemon sitting on top of your bluez/bluetooth stack, waiting for RFCOMM devices to connect. The daemon will then run scripts or daemons on that new rfcomm connection. This can be used to create a custom bluetooth printer without buying some dedicated hardware device. See the filter spirofilter in the repository for Have fun!

some pcl printer script.

Here is the first release: [1]

All questions and comments welcome!

Please send them to Christoph Lohmann <20h@r-36.net>

or come on bitreich.org IRC #bitreich-en.

[1]
gopher://bitreich.org/9/scm/rfcommd/tag/rfcommd-v0.2.tar.lz
gopher://bitreich.org/9/scm/rfcommd/tag/rfcommd-v0.2.tar.lz.sha512sum
ftp://ftp@bitreich.org/releases/rfcommd/rfcommd-v0.2.tar.lz
ftp://ftp@bitreich.org/releases/rfcommd/rfcommd-v0.2.tar.lz.sha512sum
gopher://bitreich.org/9/scm/rfcommd/tag/rfcommd-v0.2.tar.gz
gopher://bitreich.org/9/scm/rfcommd/tag/rfcommd-v0.2.tar.gz.sha512sum
ftp://ftp@bitreich.org/releases/rfcommd/rfcommd-v0.2.tar.gz

2022-03-06 GangBAN aftermaths

20h

This Sunday was a fun one. After lunch we had the supertuxkart tournament of five(!) players competing against each other on various tracks. All kind of CPUs and hardware setups participates and rushed off the cliffs.

In the evening there was the huge OpenRA battlefield. Sadly the hardware requirement of OpenRA is too

high, so only two players could participate. But this time against seven other AIs. The humans won multiple times!

See you at the next GangBAN!

Sincerely yours, 20h Chief Gaming Officer (CGO)

Breaking free from medical devices

20h

Unlike most USB gadgets around, medical devices require a specification to be proven fit for handling patients data. This makes doctor-hacking difficult for the sake of better control over what is allowed for medical use.

While this may sound as a non-starter for many, not all doctors are discouraged. Interview with 20h:

>> You are 20h, a doctor in Falken, the best village to live in in Germany, is that correct?

Yes.

>> You managed to do some hacking around a medical device. What was it? How did it help you in your diagnostics?

I wrote `rfcommd` to have my spirometer print out the results to a standard printer. It helps me having a more detailed view on the results.

The normal printout is just like 8 centimeters wide. Now it is A4.

I plan on using `rfcommd` to read out ECG data from a ECG for further analysis.

The collecting computer is a gentoo hardened on `x86_64`, with a standard bluetooth dongle, sending the print jobs via TCP/IP to a network printer.

For printing there is a cups installation, converting the PCL output of the spirometer to postscript for the network printer.

>> What software were provided to collect the data on a computer? On which kind of system was that run-

ning?

Before rfcmmnd there was no collection of the data. The spirometer has some built-in printer, which is very expensive and the printout is small.

>> Are you using it often?

I/We are using it every day for printing out spirometry (lung function) results.

By the way. A secondary function why rfcmmnd has filters: We have a sterilization device, which has a serial printout of sterilization runs.

This is what rfcmmnd does print out too.

The features of rfcmmnd moved from: Accept every rfcmm request to having filters per device mac, was because of those two devices.

But it will allow to have the ecg read-out as a filter for free.

>> It had limited interaction, and yet you managed to make it available from a linux computer. How did you do it?

First I had a python script using pybluez to offer some bluetooth printer service, which bluetooth clients connect to and send print jobs.

But I migrated this to some C implementation and generalized it as rfcmmnd so it is more modular for me and others can reuse it too.

Bluez stack had some rfcmm client application, but it was removed in newer version because they hate commandline users.

>> Was it difficult? How long did it take?

Digging around bluetooth is difficult. It looks similar to TCP/IP, but is its own terminology, protocols and principles. Look at rfcmmnd for how to announce some service.

It took me two weekends to write rfcmmnd as it is now.

>> What would you advise to designers of such devices to make everyone's life easier?

If you mean medical devices: Please open source all firmware and open up all schematics. In ten years you will be dead or in pension but still people can extend or update your devices.

And second: Never have specific assumptions and fool end users into costly standard. You never know better than your users.

For example in the spirometry description, they say, that only some bluetooth printers are compatible.

This is due to the bluetooth standard not having defined, **what** is sent to bluetooth printers.

It should be the minimum, to define this, as it is in the USB printing standard.

>> What kind of protocol interface would have been the easiest?

The easiest protocol interface, also considering security and data protection standard, would be ssh over TCP/IP. Everyone knows SSH, it can be integrated into everything and it is easily upgradable to newer security standards.

>> What does it permit to do that was not possible before?

With the spirometry data ready as simple text data, I can further process it using standard unix tools, in case I ever need this.

>> Are other people using it in the practice as well? Even indirectly?

My nurses use it mainly. They press the »print« button on the spirometry device and it prints the results.

I, as doctor, only see the printed out results and explain them to patients.

>> Does she have to use command line interface for that?

No, it's all practical. The spirometer starts its bluetooth client for `rfcomm` and `rfcomm` runs the `spirofilter` printing filter script, which invokes `lpr(1)`.

>> Are there many situations like that, where cumbersome interfaces makes life harder for working with medical devices?

Yes, it's built into all medical devices to enforce proprietary and expensive Windows software to be bought.

For example the newer version of my ECG device has some undocumented network mode. The ECG standard I will be using over serial was defined in 1990. Since then old devices only got bluetooth and ethernet, but did nothing else new.

The price stayed the same, of course.

>> Do you think designers would benefit themselves from offering another interface that is easier to use?

In the short term viewpoint it protects you from competitors to enter the market. But in the long run, this now stops me from easily processing patient data for further research. I am using a 25 yr old ECG and some 10 yr old spirometer.

>> Are there any similarities in other devices to reuse the existing work you just did?

Yes. Bluetooth is the new hype in medical devices. All those smart devices for body measurement are for example BLE, some insecure bluetooth standard to read out `key=value` from bluetooth clients. Some `bled(8)` should be easy to write.

Nearly every medical device still has some serial port, either for communication or measurement.

For measurement this will never die out, since raw data is required.

And some `serial2bluetooth`, that's what I am using for my practical examples.

>> Would it have been possible to build such device yourself from parts, but with sane interfaces instead?

Building such a device is not the hard part. The hard part is licensing the device as being a medical device.

I am, as a doctor, am allowed to license some medical device for my patients. But if I'd want to sell or give this device to some other doctor, I'd need some EU medical device license.

This is a complex process.

You have several medical device classes. Some always require some EU-wide licensing.

The logic of some ECG is very simple. But licensing it for selling is what makes it expensive and/or keeps the competition low.

It's the same for software. If you can't write it on your own, force them to open up standards, because you want to extend the software.

For extension of software, reverse engineering is legal.

>> What do you advise to people also stuck with cumbersome device, but without reverse engineer superpowers?

Force the device producers to open up standards. Write into contracts, that devices have to be interoperable, so producers need to adapt.

Carrying the Cross

tgtimes

Walking on the streets, slowly, slowed-down by carrying a huge wooden cross, tall as three persons, painted in blue, a recognisable cross shaped as an 'f', the 'f' of facebook.

This is the project Filipe Vilas-Boas, inviting anyone to watch the unrealistic scene, and question themselves on the weight of social media, and beliefs associated with technology.

>> investigating global interconnection utopia, spiritual magic and contemporary algorithmic slavery dystopia

Was there an event declaring that technology was not only for looneys on their geek basement? The opening

of facebook? The advent of the iPhone? The first day you could be fired from an office job for not being able to turn on a computer? Technology did not really appear all at once in our lives, and does not even reach every citizen of every country. Looking at ourselves with a fresh candid look and wondering if how we live make sense is becoming increasingly difficult.

Like Filipe Vilas-Boas, artists offer us a tiny window onto our own life, a porthole toward ourselves, for allowing us to watch ourselves from the outside. [1]

[1]
<https://filipevilasboas.com/Carrying-The-Cross>

Quoting Ganssle in The Embedded Muse mailing list:

>> The University of Maryland's Ralph compiler would abort after 50 compiletime errors and print out a

picture of Alfred E. Neuman, with the caption "This man never worries, but from the look of your code, you should." [1]

[1]
<http://www.ganssle.com/tem/tem439.html>

High-Tech, Low-Life

High-Tech Refers to the ability to use complex tools created by engineering, or hacking things together.

Low-Life Refers to those put aside by society, such as criminal or drug dealer, making itself edgy; or hobos and beggars, pushed to the edge by more or less everyone.

One way to develop the idea of High-Tech Low-Life would be a criminal using modern tools such to empower its crimes. A transaction giving the bad guys the big guns. Not good.

But another way to portray it is someone rejected by its surroundings, seeking support through technological tools. May it be as a source of direct income, or as a way to get informed, or inform its surrounding, perhaps the entire world such as what did happen with the late revolts in China.

The "High Tech, Low Life" (2012) documentary shows us that it is not a science-fiction plot, but a phe-

nomenon happening today.

Giving High-Tech toys to poor population sounds more like a GAFAM (Google, Amazon, Facebook, Apple, Microsoft) plan to rule over the third-world while looking like a humanitarian hero saving the world. But another way to see it is surrendering the Low-Life people to the claws of High-Tech corps, extending further the frontiers of ad-tech.

Giving entertainment platform is probably not the most urgent kind of technology people without a meal a day is going to need. What about a tractor though? In its simplest form, in China again, a 55 years-old lady farmer started to use a hoverboard (board to stand on with a wheel on left and right) to change 3 hours of daily walk to carry the vegetables harvested, into 40 minutes riding this board. [1]

Or what about deploying long-range point-to-point wireless links in west Africa to circumvent the poor cable

infrastructure? This would help escaping the lobby and regulations that take over the few IT resources of that country? [2]

Or even inventing affordable small solar or wind-power stations for the tight budgets of off-grid villages? Or an on-street display continuously showing live job offers?

[1]
<https://nextshark.com/chinese-farmer-hoverboard-life/>
<https://www.chinanews.com.cn/tp/hd2011/2018/02-13/800254.shtml>

[2]
<http://www.melissadensmore.com/papers/m4d08-mho-reassessing.pdf>
<https://www.resilience.org/stories/2015-10-27/how-to-build-a-low-tech-internet/>

FreeDOOMDay on 2022-03-27

20h

In comemoration of the beginning summer time in central Europe, we will celebrate FreeDOOMDay! On 2022-03-27 20:00 CEST (be careful!), we will play chocolate-doom [1]

This is a doom variant which runs on nearly every machine out there and supports extra modes: [2]

[1]
<https://www.chocolate-doom.org>

[2]
https://www.chocolate-doom.org/wiki/index.php/Three_screen_mode

>> Did you open-source a driver for the community as part of your job? Installed Linux on an old laptop for someone in need? Convincing the boss to make the project open-source? Attended a surprising situation of that kind? Tell us your story of High-Tech given to Low-Life on #bitreich-en IRC channel on the irc.bitreich.org server.

Please try to install the FreeDOOM wad files as a base:

See you on Sunday!

Sincerely yours, 20h Chief Gaming Officer (CGO)

Beerware: Hardware for Beer

tgtimes

Retreated industrial robot hardware recycled into a bartender. Such is the project of the Bistromatik, born in Brittany, now visiting countries

abroad.

A mechanical robot arm was built for the industry, but while still working, was removed from production, and

collected dust in a warehouse.

Jean-Marie Ollivier took this bored machine that he named "Nestor", got it to move again, and rather than servicing the industry, was programmed it to serve beers.

>> It is not rare to see Jean-Marie make Nestor dance on a violin melody.

Moving from town to town, this iron giant, taller than any human, goes on display grabbing gobelets, filling them at the tap, and offering them to the curious crowd passing by.

And if you feel hungry too, you may ask it for a treat, it can also prepare some *crepes*, the Bretons' favorite dessert. [1]

[1]
<https://bistromatik.com/>

Memecache atom feed

20h

Thanks to the innovation from the Netherlands, we can now offer an atom feed for the memecache at bitreich.org: [1]

Please subscribe for your newest meme pleasure!

Sincerely yours, 20h Chief Meme Officer (CMO)

[1]
gopher://bitreich.org/0/memecache/news.atom

St-Lazare's Paris Train Station

tgtimes

Ah! The *Saint Lazare* train station. Emblem of the Parisian train station, and today still looking like on the painting by the XIXth century painter Monet.

This typical look were somehow preserved regardless of the modernisation of the train equipments. Lately, new equipments have been installed to prevent fraud: ticket barriers are now surrounding all the stations and

their surrounding, only letting those owning a ticket onto the station.

Not unexpected from a train company for a country with fraud around 10% on long train lines. Mr. Monet would probably still be able to come and settle down for painting the train station nowadays, although to the price of a ticket to anywhere.

Yet the devices themselves seem not of the greatest comfort to both fraudsters, beggars frequently coming where most passengers are, and legitimate passengers alike. While it might be improved shortly, there is a high error rate for passengers trying to insert their ticket or NFC card.

In case of a misunderstanding of how to use these devices, the train stations are not overcrowded with staff to welcome passengers in need for in-

formation, and it would take a bit of time.

Setting up a new solution seems a difficult challenge, putting in compromise price to setup, comfort of use, reliability, finding the new staff in charge of maintenance... A reminder that technical solutions only solve technical problems. [1]

[1]
<https://lenouvelautomobiliste.fr/actualites/39949/des-portes-pour-transformer-la-vie->

FreeDOOMDay results

20h

Thanks to everyone participating in our first tryout to play doom over our bitreich infrastructure. It worked out pretty well. In the end we played the freedm.wad of freedom.

Some statistics: Maximum up and down bandwidth required was 14 kbytes/s. Maximum CPU usage here: 2% of one core. RAM: 400 kb.

Chocolate Doom is compatible to vanilla doom. Everyone having some old DOS doom can join in using rfcommd: [1]

Just attach a serial2bluetooth dongle and some bluetooth dongle in your linux machine, then use the new added filter: [2]

This will automatically connect your serial connection to a doom server over tcp/ip. Change it to bitreich.org and the standard port and you are set.

Of course you can use socat from some ttyUSB0 or ttyS0 too. Nothing stops you, but your own laziness. The possibilities are endless.

See you next time, with whatever machine you can find and which runs DOOM!

Sincerely yours, 20h Chief Gaming Officer (CGO)

[1]
`git://bitreich.org/rfcommd`

[2]
`gopher://bitreich.org/1/scm/rfcommd/commit/9b77ca90e9cf4ca7cd9521e6756dc2b833cdefce.gph`

What can possibly go wrong while sending a device entirely controlled by software on a remote location where noone would ever be able to go for a long while? The question opens a vast field of answers.

1997, Pathfinder, a solar-powered ground lander and station, with VxWorks proprietary real time operating system onboard, embedding an 6-wheeled Sojourner rover with custom firmware, landed on Mars.

During a field data collection mission a priority inversion did happen on the Pathfinder station total loss of control for the time of a reboot.

The bug was reproduced on earth and patched, latter explained on a mailing list, published online. [1]

At its core, most operating systems are built around a scheduler that orchestrates execution of many tasks onto one or several CPUs. It is a critical piece of software in the case of real-time operating systems, that must ensure to trigger some actions right on time.

Complex systems may be unfit for such purposes, and software simplicity has found its way through experimenting how complex systems may

end-up in difficult-to-debug situations.

Imagine yourself in charge of reproducing a bug on earth for something that went wrong on another planet, with a patch expected for next Monday. A strong argument toward keeping systems simple and easier to debug.

Although, the Mars operating system landscape is not all VxWorks and nothing else. For instance, the RTEMS system, Real-Time Executive for Multiprocessor Systems was open-sourced from US army 1993 and is today actively maintained by both corporations and the open source community.

Being part of Google Summer of Code, it is also welcoming newcomers to real-time operating system development, who might be able to contribute to embedded software making its way onto space. [2]

While the ISS project was put at threat by the current events in Ukraine involving all nations, outterspace still represents a middle ground where all sides have a same objective and can collaborate: extending the horizons above what could be reached before.

[1]
https://www.cs.unc.edu/~anderson/teach/comp790/papers/mars_pathfinder_long_version.htm

[2]
<https://www.rtems.org/>

The National Institute of Health is well used to the Gopher protocol, for it used it as a way to publish medical documentation. You named it: *PubMed* itself have been delivering documents through Gopher:

Phone books with name, phone number and e-mail addresses of those willing to submit it,

Images like weathermaps,

Audio such as 1992 presidential debates,

Books and all kind of publications, also proposed to users as a way to publish their own content,

Videos short ones, but also on-demand movies!

Telnet interfaces with login and password,

Search engines For browsing this entire content.

The technical bulletin of March-April 1994 reveals as much. While 1994 does not sound like a world gifted with nowadays unlimited technology, equivalents to modern tools, with less bells and less whistles, were already widespread among providers, but much less used as they are today:

Spotify were files through Gopher.

Netflix were files through Gopher.

PubMed, ResearchGate were files through Gopher.

Instagram were files through Gopher.

Facebook were publication as files through Gopher.

Amazon Kindle were text files through Gopher.

Office365 were telnet interactive session, or WordStar, PostScript, and ASCII files through Gopher.

Google was either gopher search, or interactive telnet sessions, with sometimes powerful query languages, permitting to filter the result held in the databases: *Searching for references about Italians with AIDS that are not indexed with ITALY (MH)*

This showcases that a lot of things declared as *possible today thank to the advances of technology* were available since as early as 1994. With much less bells and much less whistles. With much less bandwidth for everyone, but existing bandwidth much less used as well.

Interactive database querying languages would look a bit uninviting, and TurboGopher (showcased in the document) has not all the font, layout, media integration features of modern day web browsers.

Under that perspective, the race to technology looks like not a quest for new use-cases, but taking what was possible in the early days to in a crude format and only to some initiated, to the masses, in an inviting lay-

out, packed onto small, shiny objects that fit on a mere pocket. [1]

One year later, the Gopher for Science and Medecine project still is blown at full steam, as the National Library of Medecine publishes a bibliography for setting-up gopher servers for collaborating on specific medical topics.

[1]
https://www.nlm.nih.gov/pubs/techbull/archive/nlm_technical_bulletin_march_april_1994

[2]
<https://pubmed.ncbi.nlm.nih.gov/7599590/>

>> Developing a subject-specific Gopher at the National Library of Medicine [2]

Secret voting for Bitreich Council

20h

Bitreich is always ahead in its structure, organisation and technology. So is our democracy: [1]

The majority of council members has decided, that:

>> Secret voting is possible on certain topics. When council members vote in secret, they need to vote under a bedcover. Multiple council members can be under one bedcover.

Bitreich is reacting to the decision of Debian to introduce back chamber corruption in its decision making: [2]

This is completely prevented in the Bitreich model, since multiple council members are allowed under one bedcover, while hidden from any eavesdropper in the room.

Sincerely yours, 20h Chief Democracy Officer (CDO)

[1]
`gopher://bitreich.org/1/scm/bitreich-council/commit/f43daad938405d966c158a12b6fcb8f13a9d1868.gph`

[2]
<https://lwn.net/Articles/889444/>

In the style of the Phrack online resource, tmp.Out publishes its second volume.

>> TMP.0UT stands on the shoulders of giants, and we lend a hand for the next generation of giants to stand on ours.

Focused on the ELF format reverse engineering, the online zine culminates a rich set of resources and articles by experts for everyone interested in the world of ELF hacking.

- Bare Metal Jacket
- How to write a virtual machine in order to hide your viruses
- Every Boring Problem Found in eBPF

And much, much more... News straight out of the compiler: [1]

[1]
<https://tmpout.sh/2/>

Bitreich migrating to Windows Server

20h

Yesterday the last SSH.com license we had expired. We are now unable to access Linux on the old bitreich.org servers. In an approach to modernize Bitreich, the council decided to go further:

- Windows Server 2022 will be the new server OS for growing our business opportunities and fast deployment of critical workloads such as SQL Server with confidence using 48TB of memory, 64 sockets, and 2048 logical cores.
- Irc.bitreich.org will be replaced by Microsoft Teams to create a more engaging meeting experience with together mode. Focus on faces, pick up on nonverbal cues, and eas-

ily see who is talking.

- The ed(1) cloud will be replaced by Microsoft Office 365 to connect and empower every employee, from the office to the frontline worker, with a Microsoft 365 solution that enhances productivity and drives innovation.

We hope to see you on the new services, which enrich your daily business life.

Sincerely yours, 20h Chief Technology Officer (CTO)

Linux Sysadmin Job Offer

announce

The web is hiring over and over. A lot of professions were converted from something, to something with online web tools and a lot of computer systems are using a webinterfaces that are just skins for a database.

If you feel like giving a good sweep in all the dust of web servers, and transform fragile, complex, buggy ecosystems onto leaner, more stable systems, and are currently looking for a job as an Admin, we might have an offer for you.

The offer is located in France, within a warm and horsing team in a 20-sized company powering a little part of the Internet (not only the Web), dealing with clients from local shops

to international groups.

Come and discover the culture of Lille, in North of France, one of the only places where you can taste both Carbonnade (Belgian, meat cooked onto Belgian beer) and Welsh (Great Britain, quality melted cheddar served on a dish).

Contact *josuah* on #bitreich-en channel on irc.bitreich.org server to know more about it.

Publishing in The Gopher Times

you

Want your article published? Want to announce something to the Gopher world? Directly related to Gopher or

not, reach us on IRC with an article in any format, we will handle the rest.

ircs://irc.bitreich.org/#bitreich-en
gopher://bitreich.org/1/tgtimes/
git://bitreich.org/tgtimes/