

Copyright (C) 2020 mirage335
See the end of the file for license conditions.
See license.txt for metaBus license conditions.

Reference implementation - 'metaBus' intended to provide minimalist source-code examples to include in other projects.

Usage

```
# Interprets 'talk' , responds with corresponding 'talk' . Basic command line interface example.
echo 'PASS' | ./ubiquitous_bash.sh _interpreter ; echo

# Basic command interpreter - interprets 'talk' , responds with corresponding 'talk' .
# Largely serves to demonstrate a well-arranged implementation of a processing chain.
# No production use. *Strongly discouraged.* Actual applications realistically cannot assume transport assures complete
input/output strings.
# Directly piping from one command to another is possible if the '_process_reflector_*' functions own standard output is
redirected to a file (suppressing inappropriate log messages to actual standard output as suggested by comments in script
'reflector.sh').
echo 'PASS' | ./ubiquitous_bash.sh _reflector_ask
echo 'PASS' | ./ubiquitous_bash.sh _reflector_answer
echo 'PASS' | ./ubiquitous_bash.sh _reflector_present
```

Inspection

```
MetaEngine "$metaDir" directories may be examined by adding a delay (ie. 'sleep 90' ) before or overriding
'_stop_metaengine_wait' .
Diagnostic output may usually be correctly ordered by limiting parallelism, adding a delay (ie. 'sleep 3' after or overriding
'_processor_launch'.)
```

Demonstrations

Interpreter - Multi-Threaded Chain Talk Processor

```
'echo' 'PASS' '|' './ubiquitous_bash.sh' '_interpreter' '2>/dev/null' ';' 'echo'

# init: start metaengine_host
# init: set me null_in
# set: metaObjName= me_processor_interpreter_file_in
# write: me_processor_interpreter_file_in
# init: set me
# metaDir= tmp
# metaReg= tmp
# init: set me io
# valid: name_in
# init: set me io_name_in
# return: success
# return: success
# init: set me io
# valid: name_out
# init: set me io_name_out
# return: success
# return: success
# ##### SET
# metaBase= /home/commonadmin/project/_playground/metaBus
# metaObjName= _me_processor_interpreter_file_in

# metaID= bZjVWBBZ7xjJWtkY7w

# metaPath= bZjVWBBZ7xjJWtkY7w
# metaDir_tmp= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w
# metaDir_base= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w
# metaDir= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w
```

```

# metaReg= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg
# metaConfidence= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/confidence

# ##### PATH
# in_me_a_path= /dev/null
# in_me_b_path= /dev/null
# out_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/ao
# out_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/bo

# fork: _me_processor_interpreter_file_in:
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w/me.sh
# metaType=
# launch: _me_processor_interpreter_file_in
# init: _wait_metaengine
# ready: in_me_a_path, in_me_b_path
# ready: in_me_a_path, in_me_b_path
# processor: _me_processor_interpreter_file_in
# init: _start_metaengine
# init: _set_me
# metaDir= tmp
# metaReg= tmp
# init: _set_me_io
# valid: name_in
# init: _set_me_io.name_in
# return: success
# return: success
# init: _set_me_io
# valid: name_out
# init: _set_me_io.name_out
# return: success
# return: success
# ##### SET
# metaBase= /home/commonadmin/project/_playground/metaBus
# metaObjName= _me_processor_interpreter_file_in

# metaID= bZjVWBBZ7xjJWtkY7w

# metaPath= bZjVWBBZ7xjJWtkY7w
# metaDir_tmp= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w
# metaDir_base= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w
# metaDir= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w
# metaReg= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg
# metaConfidence= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/confidence

# ##### PATH
# in_me_a_path= /dev/null
# in_me_b_path= /dev/null
# out_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/ao
# out_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/bo

# init: _report_metaengine
# init: _relink_metaengine
# valid: name_in
# init: _relink_metaengine.name
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/null
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/null
# return: complete
# return: success
# init: _relink_metaengine
# valid: name_out
# init: _relink_metaengine.name
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w/ao
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/ao
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/bZjVWBBZ7xjJWtkY7w/bo
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/bo
# return: complete
# return: success
# init: _set_me.rand_out
# cycle: in_me_a_name= (out_me_a_name)01_file_in cycle: in_me_b_name= (out_me_b_name)01_file_in
# rand: out_me_a_name= jz6ZgYwBWJVgBJ69bx rand: out_me_b_name= jz6ZgYwBWJVgBJ69bx
# set: metaObjName= _me_processor_interpreter
# write: _me_processor_interpreter
# init: _set_me
# metaDir= tmp
# metaReg= tmp
# init: _set_me_io

```

```
# valid: name_in
# init: _set_me_io name_in
# return: success
# return: success
# init: _set_me_io
# valid: name_out
# init: _set_me_io name_out
# return: success
# return: success
##### SET
# metaBase= /home/commonadmin/project/_playground/metaBus
# metaObjName= _me_processor_interpreter

# metaID= 3KGyK9yyB2TyYzwXXj

# metaPath= 3KGyK9yyB2TyYzwXXj
# metaDir_tmp= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj
# metaDir_base= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj
# metaDir= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj
# metaReg= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg
# metaConfidence= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/confidence

##### PATH
# in_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/ao
# in_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/bo
# out_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/ao
# out_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/bo

# fork: me_processor_interpreter: /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj/me.sh
# metaType=
# launch:
# init: _wait metaengine
# ready: in_me_a_path, in_me_b_path
# ready: in_me_a_path, in_me_b_path
# processor: me_processor_interpreter
# init: _start_metaengine
# init: _set_me
# metaDir= tmp
# metaReg= tmp
# init: _set_me_io
# valid: name_in
# init: _set_me_io name_in
# return: success
# return: success
# init: _set_me_io
# valid: name_out
# init: _set_me_io name_out
# return: success
# return: success
##### SET
# metaBase= /home/commonadmin/project/_playground/metaBus
# metaObjName= _me_processor_interpreter

# metaID= 3KGyK9yyB2TyYzwXXj

# metaPath= 3KGyK9yyB2TyYzwXXj
# metaDir_tmp= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj
# metaDir_base= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj
# metaDir= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj
# metaReg= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg
# metaConfidence= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/confidence

##### PATH
# in_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/ao
# in_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/bo
# out_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/ao
# out_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/bo

# init: _report_metaengine
# init: _relink metaengine
# valid: name_in
# init: _relink metaengine name
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/ao
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj/ai
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/01_file_in/bo
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyK9yyB2TyYzwXXj/bi
```

```

# return: complete
# return: success
# init: _relink_metaengine
# valid: name_out
# init: _relink_metaengine_name
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyk9yyB2TyYzwXXj/ao
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/ao
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/3KGyk9yyB2TyYzwXXj/bo
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/bo
# return: complete
# return: success
# init: _interpreter_filter
# init: _set_me_rand_out
# cycle: in_me_a_name= (out_me_a_name)03_interpreter cycle: in_me_b_name= (out_me_b_name)03_interpreter
# rand: out_me_a_name= GTbkXyy7XXbgtkTVBW rand: out_me_b_name= GTbkXyy7XXbgtkTVBW
# init: _set_me_null_out
# set: metaObjName= _me_processor_interpreter_file_out
# write: _me_processor_interpreter_file_out
# init: _set_me
# metaDir= tmp
# metaReg= tmp
# init: _set_me_io
# valid: name_in
# init: _set_me_io_name_in
# return: success
# return: success
# init: _set_me_io
# valid: name_out
# init: _set_me_io_name_out
# return: success
# return: success
# ##### SET
# metaBase= /home/commonadmin/project/_playground/metaBus
# metaObjName= _me_processor_interpreter_file_out

# metaID= BYtztWWZt9gtTJ77gB

# metaPath= BYtztWWZt9gtTJ77gB
# metaDir_tmp= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB
# metaDir_base= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB
# metaDir= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB
# metaReg= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg
# metaConfidence= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/confidence

# ##### PATH
# in_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/ao
# in_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/bo
# out_me_a_path= /dev/null
# out_me_b_path= /dev/null

# fork: _me_processor_interpreter_file_out:
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB/me.sh
# metaType=
# launch: _me_processor_interpreter_file_out
# init: _wait_metaengine
# ready: in_me_a_path, in_me_b_path
# ready: in_me_a_path, in_me_b_path
# processor: _me_processor_interpreter_file_out
# init: _start_metaengine
# init: _set_me
# metaDir= tmp
# metaReg= tmp
# init: _set_me_io
# valid: name_in
# init: _set_me_io_name_in
# return: success
# return: success
# init: _set_me_io
# valid: name_out
# init: _set_me_io_name_out
# return: success
# return: success
# ##### SET
# metaBase= /home/commonadmin/project/_playground/metaBus
# metaObjName= _me_processor_interpreter_file_out

```

```
# metaID= BYtztWWZt9gtTJ77gB

# metaPath= BYtztWWZt9gtTJ77gB
# metaDir_tmp= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB
# metaDir_base= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB
# metaDir= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB
# metaReg= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg
# metaConfidence= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/confidence

# ##### PATH
# in_me_a_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/ao
# in_me_b_path= /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/bo
# out_me_a_path= /dev/null
# out_me_b_path= /dev/null

# init: _report_metaengine
# init: _relink_metaengine
# valid: name_in
# init: _relink_metaengine_name
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/ao
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB/ai
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter
# _relink_relative /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/03_interpreter/bo
/home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/BYtztWWZt9gtTJ77gB/bi
# return: complete
# return: success
# init: _relink_metaengine
# valid: name_out
# init: _relink_metaengine_name
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/null
# mkdir -p /home/commonadmin/project/_playground/metaBus/.m_7GtV3XkYxG7jt6gjt看/_reg/name/null
# return: complete
# return: success
# PASS
```

Design

Please keep all examples and associated code as 'elegant' as possible - perhaps simple and nonrepetitive - to best illustrate logic flow. Flow charts may not be adequate to supplement unnecessarily complicated code.

Sufficiently reusable functions may be ported upstream to 'ubiquitous_bash' to allow more rapid or more obvious assembly of at least some specialized types of MetaEngine processing chains.

Serialization

Any serialization format should be acceptable for sending messages across shared channels. However, these channels should be assumed 'noisy' so that multiple serialization formats may be used simultaneously.

Transports

Existing IPC transports may be used with, or independently of, MetaEngine . However, the essential benefits of ad-hoc real-time redirection, innumerable 'ports', noise/collision tolerance, native executable multiplatform compatibility (UNIX/MSW), may not be directly provided.

ZeroMQ

May not offer an MSW compatible transport layer.

' inter-process communication (Linux only and not available in any of the native ports as yet) '
<https://blog.scottlogic.com/2015/03/20/ZeroMQ-Quick-Intro.html>

'Does it requires to install wsl2 or dependency is natively available without install wsl2 ?

It is an hard dependency :(not easy to manage with installer.'
<https://github.com/zeromq/libzmq/issues/153>

Redis

May not offer an MSW compatible transport layer.

<https://redislabs.com/blog/redis-on-windows-10/>

RabbitMQ

<https://www.rabbitmq.com/tutorials/tutorial-three-python.html>

DBus

"If you use this low-level API directly, you're signing up for some pain."
<https://dbus.freedesktop.org/doc/api/html/index.html>
<https://hackaday.io/project/279-sonomkr-noise-monitoring/log/86364-zeromq-vs-dbus-for-pub-sub-pattern>

FIFO

As typically provided by OS, FIFO (named pipe) may not be directly receivable by multiple recipients.

<https://stackoverflow.com/questions/33613341/how-to-send-broadcast-messages-with-named-pipe-using-c>

TCP

A possible workaround may be to declare a 'standard' TCP port for IPC by the 'primary' user of a computer, an approach typical of 'extendedInterface'.

Distribution

Best practice may be to add a 'git submodule' to a '_lib' subdirectory of any project using 'metaBus' .

Ideally, an uncompressed, squashfs, or otherwise accessible filesystem should always be made available for developers who may need to quickly maintain associated hardware (ie. 3D printers in factories). However, good compression of source code including multiple repositories with 'metaBus' submodules might be achieved with 'bup' .

Redistribution and Preservation

Entire 'recursive' clones of the 'metaBus' and derivative repositories may be appropriate for redistribution and preservation.

```
git clone --recursive https://github.com/mirage335/metaBus
```

Distributing a 'binary' version with a separate means to obtain a 'source' copy is *strongly discouraged*.

Without relevant source code, it may become infeasible to interoperate with, or to revise, potentially expensive, complicated, or otherwise entrenched hardware and software systems. Entire point of 'metaBus' project, is to ensure new or redundant modules, may be connected ad-hoc, to existing systems. In particular, 'metaBus' is intended to further eliminate maintenance labor from the widespread use of 3D printing.

For 'development' work, the complete repository is a necessity which modern data storage and bandwidth is easily able to accommodate.

Particularly, please respect the 'Remote Network Interaction' requirements of the GNU Affero General Public License associated with this software - author's desire is that all clients must have prominent notices of 'Corresponding Source' availability. Simply because the communications happening over 'metaBus' networks may or may not be intended for *direct* end-user interaction, is not an excuse for unreasonable lack of due diligence.

As examples, the following use cases include end-user interaction which provides at least reasonable opportunity for prominent notification of 'Corresponding Source' availability.

- * Documentation, hardcopy or otherwise, intended for end-user reading, in any way distributed with 'metaBus' software, or with hardware which may include 'metaBus' software.
- * Messages sent by 3D printer firmware over serial port (eg. UART) at startup and/or as typical of responses to G-Code commands, which may include end-user notices of other copyright, firmware, machine name, etc.
- * Client programs (eg. 'pronterface') which are intended for end-user interaction and are in any way distributed with 'metaBus' software, or with hardware which may include 'metaBus' software.
- * An occupancy sensor graphic indicator on an internet accessible service intended for end-users which may include commentary, notices of other copyright/authorship/ownership, etc.

To be clear, in such cases which provide at least reasonable opportunity for prominent notification of 'Corresponding Source' availability, author's desire is to require such prominent notification.

Patent enforcement by distributors of metaBus may be subject to the requirements of the GNU Affero General Public License associated with this software. Additionally, some patent holders have pledged to limit their own enforcement of patents.

'Open Invention Network (OIN) is a company that acquires patents and licenses them royalty-free to its community members who, in turn, agree not to assert their own patents against Linux and Linux-related systems and applications.[1]'
https://en.wikipedia.org/wiki/Open_Invention_Network

'Microsoft has irrevocably pledged not to assert any claims against open source developers[21] which CEO Steve Ballmer called "an important step and significant change in how we share information about our products and technologies."[22] This pledge has been accepted with some skepticism.[23]'
https://en.wikipedia.org/wiki/Software_patents_and_free_software

'It has now yielded, under pressure from the market and from the European Commission.'
https://web.archive.org/web/20120910183540/http://www.gartner.com/DisplayDocument?doc_cd=155733
<https://www.microsoft.com/en-us/legal/intellectualproperty/ip-and-open-source>
<https://www.zdnet.com/article/microsoft-open-sources-its-entire-patent-portfolio/>

Future Work

- * Demonstrate ability to preserve less dependable SSDs (ie. typical SD Cards) by automatically detecting usable '/dev/shm' as "\$metaDir" (derived from "\$metaTmp") location (also applies to upstream Ubiquitous Bash).
- * Real-time examples using FIFO , many-to-many 'broadcast' over FIFO and/or TCP/UDP , etc .

Reference

Copyright

This file is part of metaBus.

metaBus is free software: you can redistribute it and/or modify it under the terms of the GNU Affero General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

metaBus is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Affero General Public License for more details.

You should have received a copy of the GNU Affero General Public License along with metaBus. If not, see <<http://www.gnu.org/licenses/>>.