

# Modern USB Gadget with Custom USB Functions...

...and its integration with systemd

**Andrzej Pietrasiewicz**  
<[andrzej.p@collabora.com](mailto:andrzej.p@collabora.com)>

**Linux Piter 2019**



COLLABORA

Open First

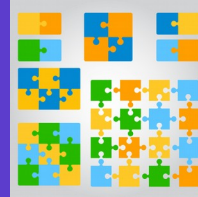
➤ **USB Device  
user?**



A Linux machine connected to a PC

# Overview

- Composition and components  
+ your own USB functionality

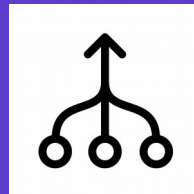


- Tools



- systemd integration

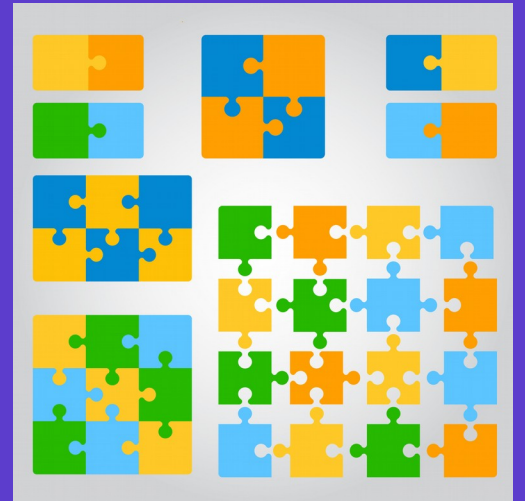
+ your own USB functionality





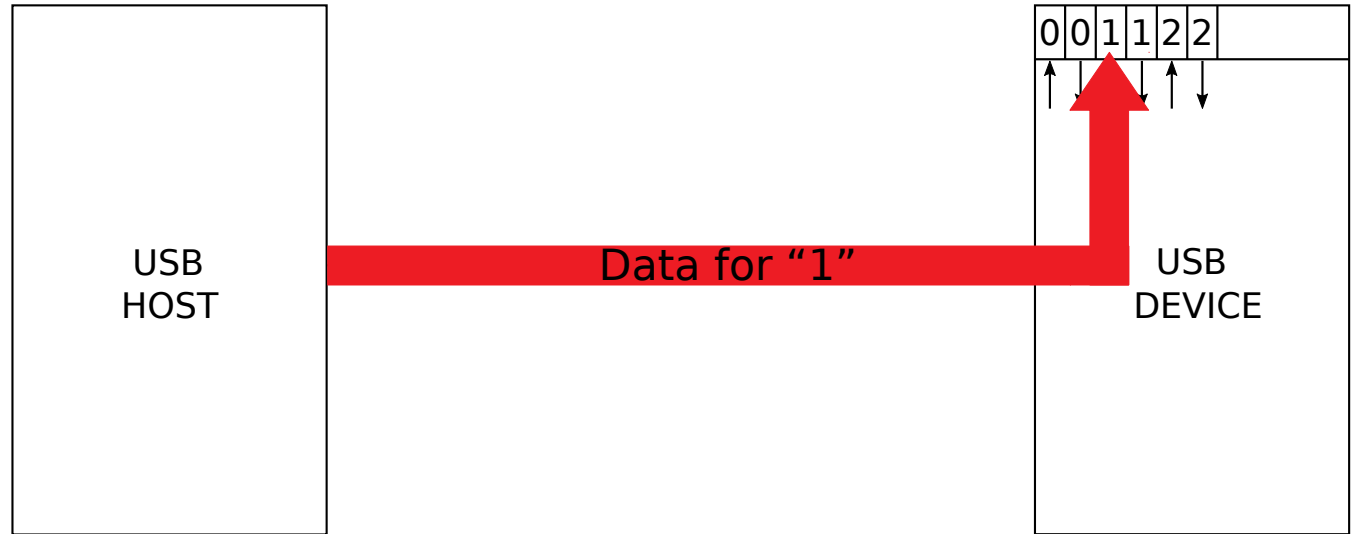
COLLABORA

# Composition and components



# Endpoints

- Control
- Bulk
- Interrupt
- Isochronous



# Interface

- Collection of endpoints for a particular purpose
- Implemented as “USB function” in Linux kernel



# Configuration

- Collection of 1+ interface(s)
- Device contains 1+ configuration(s)
  - Only 1 active at a time
  - Rarely  $> 1$



# FunctionFS

- A USB function
- A filesystem with ep0 and epX....





# Theoretical # combinations

$$\binom{20}{1} + \binom{20}{2} + \binom{20}{3} + \dots + \binom{20}{20} = \text{A LOT}$$





COLLABORA

# Tools



# Configs

- `/sys/kernel/config/usb_gadget`
- `mkdir, echo, ln -s, rmdir`

```
mkdir g1
cd g1

echo 0x1d6b > idVendor
echo 0x104 > idProduct

mkdir strings/0x409 # US English, others rarely seen
echo "Collabora" > strings/0x409/manufacturere
echo "ECM" > strings/0x409/product

mkdir configs/c.1 # dot and number mandatory

mkdir functions/ecm.usb0

ln -s functions/ecm.usb0/ configs/c.1/

# bind!
# ls /sys/class/udc to see available UDCs
echo 12480000.hsotg > UDC
```



```
mkdir g1
cd g1

echo 0x1d6b > idVendor
echo 0x104 > idProduct

mkdir strings/0x409 # US English, others rarely seen
echo "Collabora" > strings/0x409/manufacturer
echo "ECM" > strings/0x409/product

mkdir configs/c.1 # dot and number mandatory

mkdir functions/ecm.usb0

ln -s functions/ecm.usb0/ configs/c.1/

# bind!
# ls /sys/class/udc to see available UDCs
echo 12480000.hsotg > UDC
```



```
mkdir g1
cd g1

echo 0x1d6b > idVendor
echo 0x104 > idProduct

mkdir strings/0x409 # US English, others rarely seen
echo "Collabora" > strings/0x409/manufacturere
echo "ECM" > strings/0x409/product

mkdir configs/c.1 # dot and number mandatory

mkdir functions/ecm.usb0

ln -s functions/ecm.usb0/ configs/c.1/

# bind!
# ls /sys/class/udc to see available UDCs
echo 12480000.hsotg > UDC
```



```
mkdir g1
cd g1

echo 0x1d6b > idVendor
echo 0x104 > idProduct

mkdir strings/0x409 # US English, others rarely seen
echo "Collabora" > strings/0x409/manufacturer
echo "ECM" > strings/0x409/product

mkdir configs/c.1 # dot and number mandatory

mkdir functions/ecm.usb0

ln -s functions/ecm.usb0/ configs/c.1/

# bind!
# ls /sys/class/udc to see available UDCs
echo 12480000.hsotg > UDC
```



```
mkdir g1
cd g1

echo 0x1d6b > idVendor
echo 0x104 > idProduct

mkdir strings/0x409 # US English, others rarely seen
echo "Collabora" > strings/0x409/manufacturere
echo "ECM" > strings/0x409/product

mkdir configs/c.1 # dot and number mandatory

mkdir functions/ecm.usb0

ln -s functions/ecm.usb0/ configs/c.1/

# bind!
# ls /sys/class/udc to see available UDCs
echo 12480000.hsotg > UDC
```





```
mkdir g1
cd g1

echo 0x1d6b > idVendor
echo 0x104 > idProduct

mkdir strings/0x409 # US English, others rarely seen
echo "Collabora" > strings/0x409/manufacturer
echo "ECM" > strings/0x409/product

mkdir configs/c.1 # dot and number mandatory

mkdir functions/ecm.usb0

ln -s functions/ecm.usb0/ configs/c.1/

# bind!
# ls /sys/class/udc to see available UDCs
echo 12480000.hsotg > UDC
```



# gt

- [github.com/kopasiak/gt](https://github.com/kopasiak/gt)
- [github.com/libusb-gx/libusb-gx.git](https://github.com/libusb-gx/libusb-gx.git)
- gadget “schemes”

```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "ECM";
    }
);
functions :
{
    ecm_usb0 :
    {
        instance = "usb0";
        type = "ecm";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ecm.usb0";
                function = "ecm_usb0";
            } );
    } );
```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "ECM";
    }
);
functions :
{
    ecm_usb0 :
    {
        instance = "usb0";
        type = "ecm";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ecm.usb0";
                function = "ecm_usb0";
            } );
    } );
```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "ECM";
    }
);
functions :
{
    ecm_usb0 :
    {
        instance = "usb0";
        type = "ecm";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ecm.usb0";
                function = "ecm_usb0";
            } );
    } );
```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "ECM";
    }
);
functions :
{
    ecm_usb0 :
    {
        instance = "usb0";
        type = "ecm";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ecm.usb0";
                function = "ecm_usb0";
            } );
    } );
```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "ECM";
    }
);
functions :
{
    ecm_usb0 :
    {
        instance = "usb0";
        type = "ecm";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ecm.usb0";
                function = "ecm_usb0";
            } );
    } );
```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "ECM";
    }
);
functions :
{
    ecm_usb0 :
    {
        instance = "usb0";
        type = "ecm";
    };
};
```

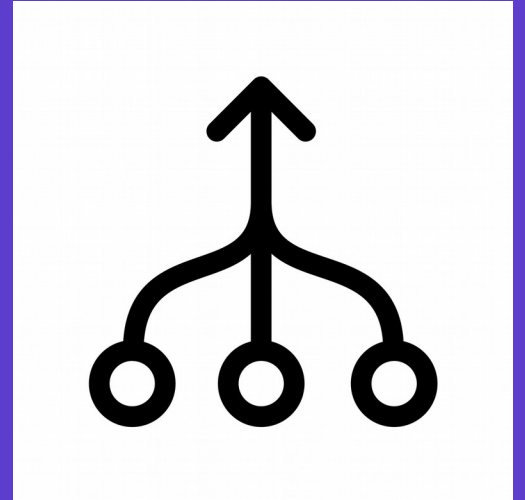
```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ecm.usb0";
                function = "ecm_usb0";
            } );
    } );
```







# systemd integration



# systemd

- Various units
  - service, mount, target, socket, ...
- Tracked dependencies
- High parallelization
- Lazy initialization (socket units)



# The “glue layer”

- usb-gadget.target
- udev event



# The “glue layer”

- usb-gadget.target

```
[Unit]
Description=Hardware activated USB gadget
Documentation=man:systemd.special(7)
```

- udev event



# The “glue layer”

- usb-gadget.target

```
[Unit]
Description=Hardware activated USB gadget
Documentation=man:systemd.special(7)
```

- udev event

```
SUBSYSTEM=="udc", ACTION=="add", TAG+="systemd", \
    ENV{SYSTEMD_WANTS}+="usb-gadget.target"
```



# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```



# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```



# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```





# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```



# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```



# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```

```
systemctl [en|dis]able usb-gadget.service
```



# Service unit

```
[Unit]
Description=Load USB gadget scheme for ecm
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load ecm.scheme ecm
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf ecm
Type=simple

[Install]
WantedBy=usb-gadget.target
```

```
systemctl [en|dis]able usb-gadget.service
```



# Service unit templated

```
[Unit]
Description=Load USB gadget scheme for %i
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load %i.scheme %i
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf %i
Type=simple

[Install]
WantedBy=usb-gadget.target
```



# Service unit templated

```
[Unit]
Description=Load USB gadget scheme for %i
Requires=sys-kernel-config.mount
After=sys-kernel-config.mount
DefaultDependencies=no

[Service]
ExecStart=/bin/gt load %i.scheme %i
RemainAfterExit=yes
ExecStop=/bin/gt rm -rf %i
Type=simple

[Install]
WantedBy=usb-gadget.target
```

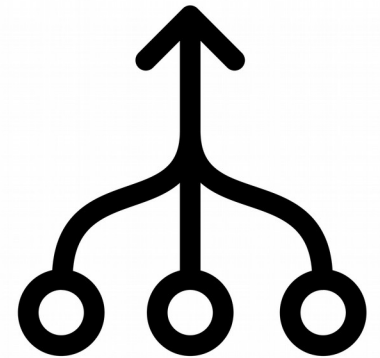
```
systemctl [en|dis]able usb-gadget@ecm.service
```





# systemd integration

Your own USB function



# FunctionFS vs systemd

- Gadget scheme
- Service unit (create gadget, gt load **-o**)
- Mount unit (FunctionFS instance)
- Socket unit (start its service unit, enable the gadget)
- Service unit (spawn userspace daemon)





```

attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "MTP Gadget";
    }
);
functions :
{
    ffs_mtp :
    {
        instance = "mtp";
        type = "ffs";
    };
};

configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ffs.mtp";
                function = "ffs_mtp";
            } );
    } );

```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "MTP Gadget";
    }
);
functions :
{
    ffs_mtp :
    {
        instance = "mtp";
        type = "ffs";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ffs.mtp";
                function = "ffs_mtp";
            } );
    } );
```



```
attrs :
{
    idVendor = 0x1D6B;
    idProduct = 0x104;
};
strings = (
    {
        lang = 0x409;
        manufacturer = "Collabora";
        product = "MTP Gadget";
    }
);
functions :
{
    ffs_mtp :
    {
        instance = "mtp";
        type = "ffs";
    };
};
```

```
configs = (
    {
        id = 1;
        name = "c";
        functions = (
            {
                name = "ffs.mtp";
                function = "ffs_mtp";
            } );
    } );
```



# Mount unit

```
[Unit]
Description=FunctionFS instance for ffs_mtp
Requires=usb-gadget-ffs@ffs_mtp.service
After=usb-gadget-ffs@ffs_mtp.service
Before=ffs@ffs_mtp.socket

[Mount]
What=mtp
Where=/run/ffs_mtp
Type=functionfs
Options=defaults
TimeoutSec=5
```



# Mount unit

```
[Unit]
Description=FunctionFS instance for ffs_mtp
Requires=usb-gadget-ffs@ffs_mtp.service
After=usb-gadget-ffs@ffs_mtp.service
Before=ffs@ffs_mtp.socket
```

```
[Mount]
What=mtp
Where=/run/ffs_mtp
Type=functionfs
Options=defaults
TimeoutSec=5
```



# Mount unit

```
[Unit]
Description=FunctionFS instance for ffs_mtp
Requires=usb-gadget-ffs@ffs_mtp.service
After=usb-gadget-ffs@ffs_mtp.service
Before=ffs@ffs_mtp.socket

[Mount]
What=mtp
Where=/run/ffs_mtp
Type=functionfs
Options=defaults
TimeoutSec=5
```



# Mount unit

```
[Unit]
Description=FunctionFS instance for ffs_mtp
Requires=usb-gadget-ffs@ffs_mtp.service
After=usb-gadget-ffs@ffs_mtp.service
Before=ffs@ffs_mtp.socket

[Mount]
What=mtp
Where=/run/ffs_mtp
Type=functionfs
Options=defaults
TimeoutSec=5
```



# Socket unit

```
[Unit]
Description=USB FunctionFS socket for %i
Requires=run-%i.mount
After=run-%i.mount
DefaultDependencies=no

[Socket]
ListenUSBFunction=/run/%i
Service=%i.service
ExecStartPost=/bin/gt enable %i
TriggerLimitIntervalSec=0

[Install]
WantedBy=usb-gadget.target
```





# Socket unit

```
[Unit]
Description=USB FunctionFS socket for %i
Requires=run-%i.mount
After=run-%i.mount
DefaultDependencies=no

[Socket]
ListenUSBFunction=/run/%i
Service=%i.service
ExecStartPost=/bin/gt enable %i
TriggerLimitIntervalSec=0

[Install]
WantedBy=usb-gadget.target
```



# Socket unit

```
[Unit]
Description=USB FunctionFS socket for %i
Requires=run-%i.mount
After=run-%i.mount
DefaultDependencies=no

[Socket]
ListenUSBFunction=/run/%i
Service=%i.service
ExecStartPost=/bin/gt enable %i
TriggerLimitIntervalSec=0

[Install]
WantedBy=usb-gadget.target
```



# Socket unit

```
[Unit]
Description=USB FunctionFS socket for %i
Requires=run-%i.mount
After=run-%i.mount
DefaultDependencies=no

[Socket]
ListenUSBFunction=/run/%i
Service=%i.service
ExecStartPost=/bin/gt enable %i
TriggerLimitIntervalSec=0

[Install]
WantedBy=usb-gadget.target
```



# Socket unit

```
[Unit]
Description=USB FunctionFS socket for %i
Requires=run-%i.mount
After=run-%i.mount
DefaultDependencies=no

[Socket]
ListenUSBFunction=/run/%i
Service=%i.service
ExecStartPost=/bin/gt enable %i
TriggerLimitIntervalSec=0

[Install]
WantedBy=usb-gadget.target
```



# Socket unit

```
[Unit]
Description=USB FunctionFS socket for %i
Requires=run-%i.mount
After=run-%i.mount
DefaultDependencies=no

[Socket]
ListenUSBFunction=/run/%i
Service=%i.service
ExecStartPost=/bin/gt enable %i
TriggerLimitIntervalSec=0

[Install]
WantedBy=usb-gadget.target
```



# Service unit

```
[Unit]
Description=MTP responder
StartLimitIntervalSec=0

[Service]
Type=simple
ExecStart=/usr/bin/cmtmp-responder
USBFunctionDescriptors=/etc/cmtmp-responder/descs
USBFunctionStrings=/etc/cmtmp-responder/strs
KillMode=process
RestartSec=3
Restart=on-failure
```



# Service unit

```
[Unit]
Description=MTP responder
StartLimitIntervalSec=0

[Service]
Type=simple
ExecStart=/usr/bin/cmtp-responder
USBFunctionDescriptors=/etc/cmtp-responder/descs
USBFunctionStrings=/etc/cmtp-responder/strs
KillMode=process
RestartSec=3
Restart=on-failure
```



# Service unit

```
[Unit]
Description=MTP responder
StartLimitIntervalSec=0

[Service]
Type=simple
ExecStart=/usr/bin/cmtmp-responder
USBFunctionDescriptors=/etc/cmtmp-responder/descs
USBFunctionStrings=/etc/cmtmp-responder/strs
KillMode=process
RestartSec=3
Restart=on-failure
```





# Service unit

```
[Unit]
Description=MTP responder
StartLimitIntervalSec=0
```

```
[Service]
Type=simple
ExecStart=/usr/bin/cmtmp-responder
USBFunctionDescriptors=/etc/cmtmp-responder/descs
USBFunctionStrings=/etc/cmtmp-responder/strs
KillMode=process
RestartSec=3
Restart=on-failure
```

include/uapi/linux/usb/functionfs.h



# Descriptors and strings

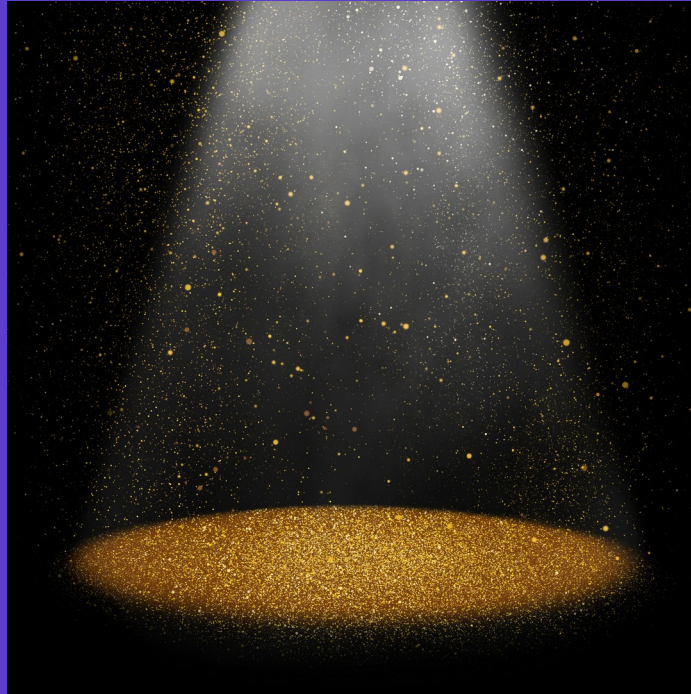
- [github.com/kopasiak/gt/pull/8](https://github.com/kopasiak/gt/pull/8)
- Declarative config → blob
  - USBFunctionDescriptors, USBFunctionStrings





COLLABORA

# Demo



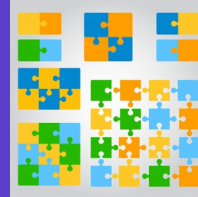
# MTP Device

- [github.com/cmtp-responder/cmtp-responder.git](https://github.com/cmtp-responder/cmtp-responder.git)
- Odroid U2



# Summary

- Composition and components  
+ your own USB functionality

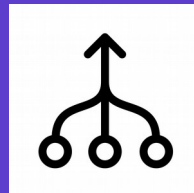


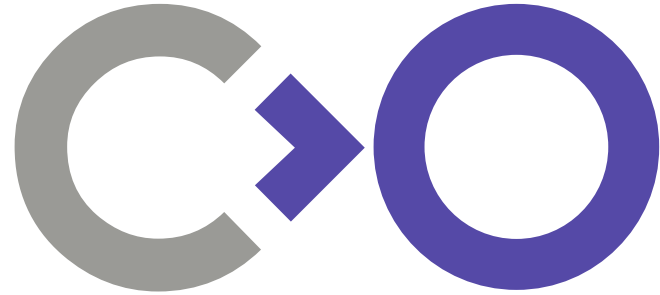
- Tools



- systemd integration

+ your own USB functionality





**Thank you!**

**<http://col.la/careers>**



COLLABORA

**Open First**