GENIVI®

AMM OPEN DAYS

How to work with GENIVI projects

October, 2015 | GENIVI AMM Korea

Jeremiah C. Foster
GENIVI Community Manager
Pelagicore AB

AGENDA

What makes GENIVI Open Source projects successful?

How do I contribute to GENIVI projects?

What are the steps to contribute to GENIVI code projects?

What are the results of contribution?

Q&A

SUCCESS

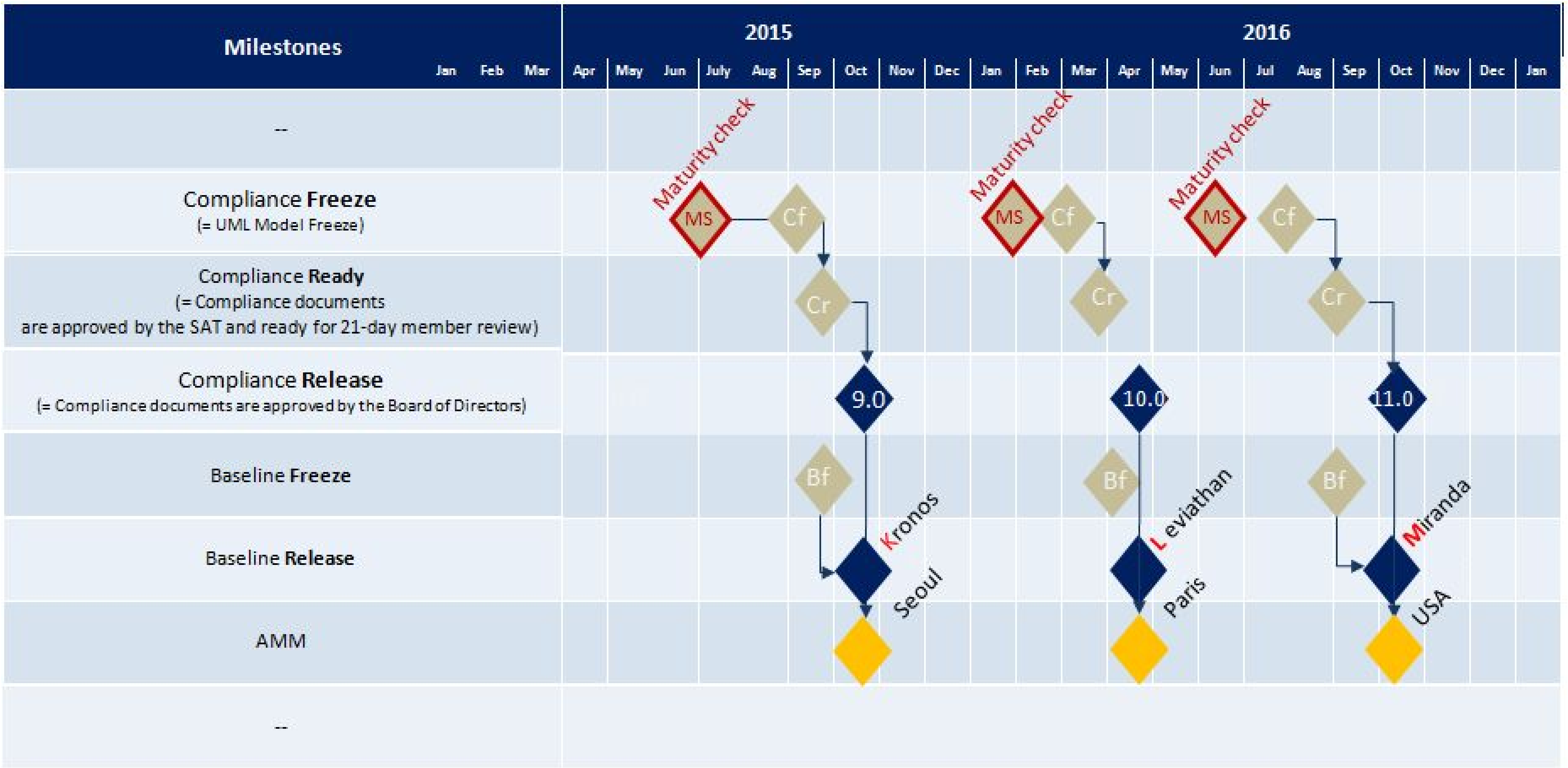
Everything that Clause-Peter said, plus;

The project addresses a real need with usable code, preferably “rock solid” production code

The project builds into (at least) GENIVI’s baseline

The project has a path into GENIVI’s GDP

RESULTS



Milestones

--

Compliance **Freeze**
(= UML Model Freeze)

Compliance **Ready**
(= Compliance documents
are approved by the SAT and ready for 21-day member review)

Compliance **Release**
(= Compliance documents are approved by the Board of Directors)

Baseline **Freeze**

Baseline **Release**

AMM

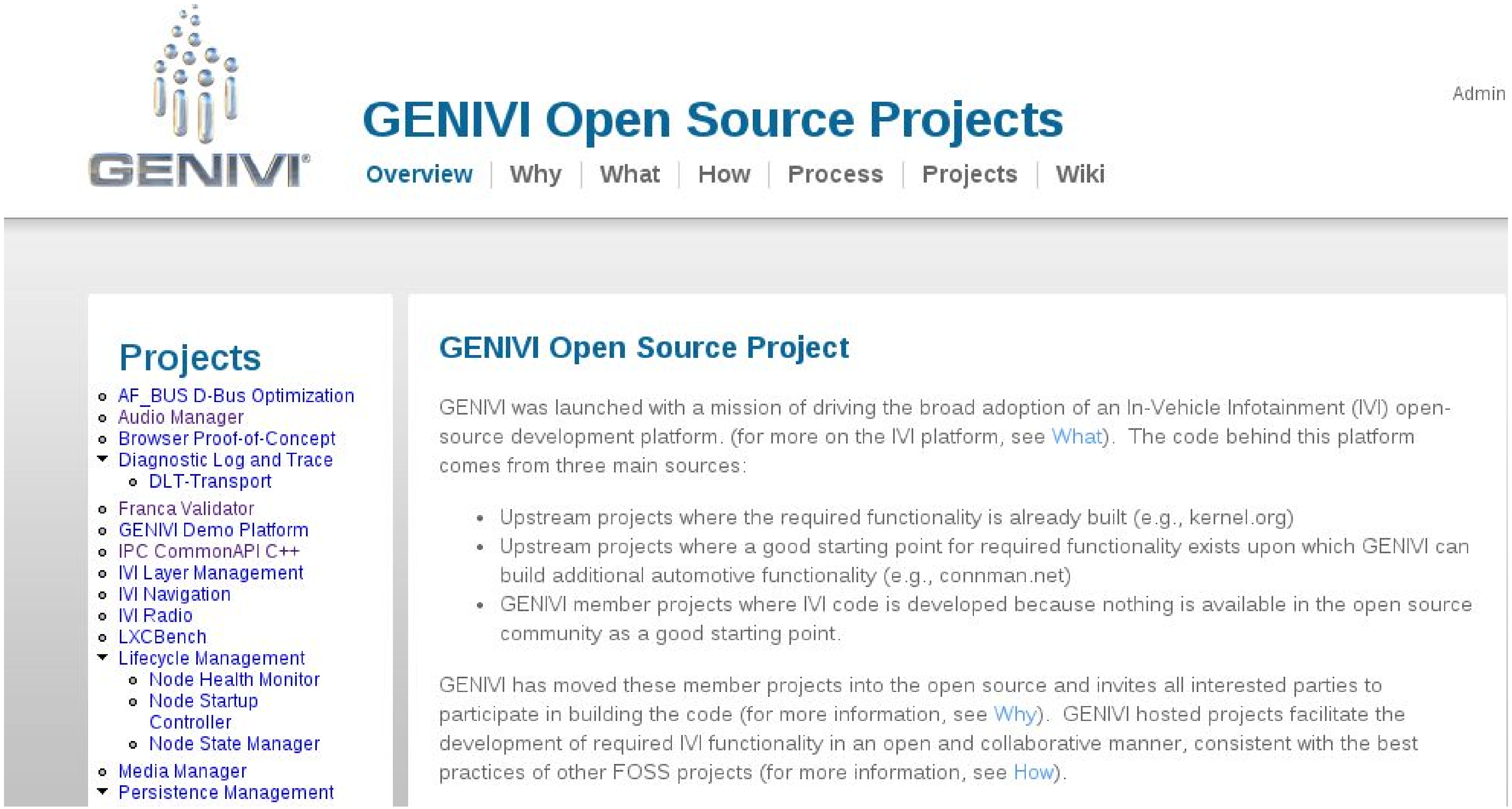
--

SIMPLE STEPS

1) “Select the code to contribute and obtain (internal) approval(s)”

- ➔ GENIVI hosts all of its source code on a public git
- ➔ http://git.projects.genivi.org

PROJECT PAGES



GENIVI Open Source Projects

Projects

- AF_BUS D-Bus Optimization
- Audio Manager
- Browser Proof-of-Concept
- Diagnostic Log and Trace
  - DLT-Transport
- Franca Validator
- GENIVI Demo Platform
- IPC CommonAPI C++
- IVI Layer Management
- IVI Navigation
- IVI Radio
- LXCBench
- Lifecycle Management
  - Node Health Monitor
  - Node Startup Controller
  - Node State Manager
- Media Manager
- Persistence Management

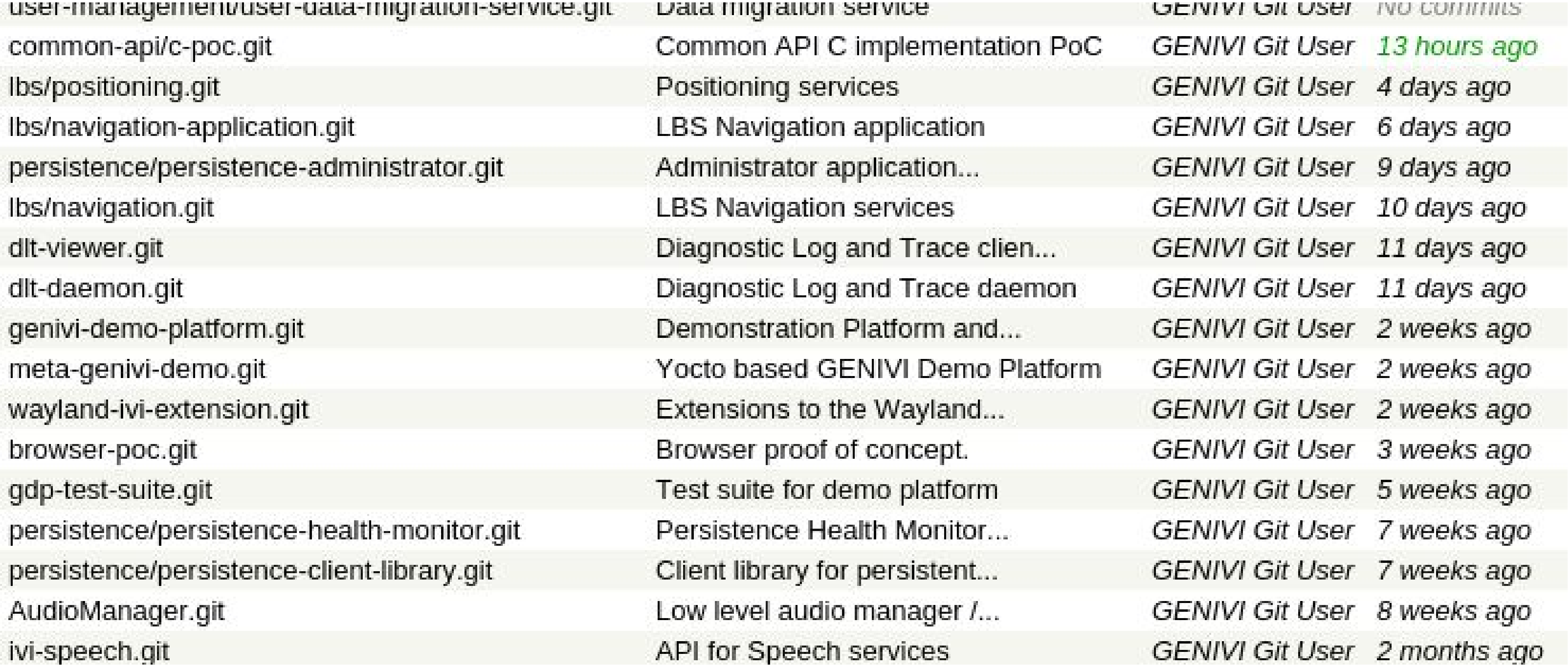
GENIVI Open Source Project

GENIVI was launched with a mission of driving the broad adoption of an In-Vehicle Infotainment (IVI) open-source development platform. (for more on the IVI platform, see What). The code behind this platform comes from three main sources:

- Upstream projects where the required functionality is already built (e.g., kernel.org)
- Upstream projects where a good starting point for required functionality exists upon which GENIVI can build additional automotive functionality (e.g., connman.net)
- GENIVI member projects where IVI code is developed because nothing is available in the open source community as a good starting point.

GENIVI has moved these member projects into the open source and invites all interested parties to participate in building the code (for more information, see Why). GENIVI hosted projects facilitate the development of required IVI functionality in an open and collaborative manner, consistent with the best practices of other FOSS projects (for more information, see How).

OPEN SOURCE IVI PROJECTS



user-management/user-data-migration-service.git	Data migration service	GENIVI Git User	No commits
common-api/c-poc.git	Common API C implementation PoC	GENIVI Git User	13 hours ago
lbs/positioning.git	Positioning services	GENIVI Git User	4 days ago
lbs/navigation-application.git	LBS Navigation application	GENIVI Git User	6 days ago
persistence/persistence-administrator.git	Administrator application...	GENIVI Git User	9 days ago
lbs/navigation.git	LBS Navigation services	GENIVI Git User	10 days ago
dlt-viewer.git	Diagnostic Log and Trace clien...	GENIVI Git User	11 days ago
dlt-daemon.git	Diagnostic Log and Trace daemon	GENIVI Git User	11 days ago
genivi-demo-platform.git	Demonstration Platform and...	GENIVI Git User	2 weeks ago
meta-genivi-demo.git	Yocto based GENIVI Demo Platform	GENIVI Git User	2 weeks ago
wayland-ivi-extension.git	Extensions to the Wayland...	GENIVI Git User	2 weeks ago
browser-poc.git	Browser proof of concept.	GENIVI Git User	3 weeks ago
gdp-test-suite.git	Test suite for demo platform	GENIVI Git User	5 weeks ago
persistence/persistence-health-monitor.git	Persistence Health Monitor...	GENIVI Git User	7 weeks ago
persistence/persistence-client-library.git	Client library for persistent...	GENIVI Git User	7 weeks ago
AudioManager.git	Low level audio manager /...	GENIVI Git User	8 weeks ago
ivi-speech.git	API for Speech services	GENIVI Git User	2 months ago

COMPONENT CHOICE STRATEGY

➔ Identify existing components that are relevant for your needs

➔ Align on APIs which are strategic to your business

GENIVI DEMO PLATFORM

- Holds nearly all of the GENIVI components including dependencies and subsystems
- Evolving into a complete SDK or ADK
- A good starting point for the latest code
- Supported by multiple hardware vendors

GETTING COMPONENT CODE

```
$ git clone http://git.projects.genivi.org/dlt-daemon.git

. . .

$ git commit -am "Fixed magic code"
```

➔ Create a patch

CONTRIBUTING TO AUDIO MANAGER

== Working on the code & contribution

.First get the code from the git:

```
git clone
```

.Get an overview of all branches:

```
git branch
```

.Switch to the branch you want to work on (see versioning scheme, the master is the feature branch) and verify that it has switched (* changed)

```
git checkout <your branch>
git branch
```

.Best practice is to create a local branch based on the current branch:

```
git branch working_branch
```

Start working, best practice is to commit smaller, compilable peaced during the work that makes it easier to handle later on.

If you want to commit you changes, send them to the audiomanager-dev list, you can create a patch like this:

```
git format-patch working_branch <your branch>
```

This creates a set of patches that are published via the mailing list.The patches will be discussed and then merged & uploaded on the git by the maintainer.

Patches can be accepted either under GENIVI Cla or MPL 2.0 (see section License). Please be sure that the signed-off-by is set correctly. For more, check out http://gerrit.googlecode.com/svn/documentation/2.0/user-signedoffby.html

COMMUNICATE

- Important to remember that GENIVI software developers are distributed throughout the world, North America, Europe, Asia
- Requires a distributed communication channel that is asynchronous, that channel is email quite often
- The mailing list serves as a database of searchable discussion that aggregates by subject matter
- Use the mailing lists!
- It is in the open, so mailing lists are public
- English is not everyone’s first language and that is totally fine

TRACKING BUGS

- GENIVI has a public bug tracker which holds bugs for GENIVI components as well as for the GENIVI Demo platform
- Easy to sign up to the bug tracker and file bugs, add patches, triage bugs
- Sometimes it can be good to discuss the issue on the mailing list first to see if it is a known issue or a local issue

GENIVI

CHECKLIST

Create git repo authorization

GENIVI allows only those with public ssh keys submitted to the server to push code to GENIVI repositories so a **public** ssh key will need to be supplied to the Community Manager or GENIVI IT manager for placement on the hosting server.

Ensure LICENSE file is included

A license is required for each code repository to designate the license the code is under. See the LRT GENIVI internal license know-how and GENIVI's GENIVI License Policy for more info.

Ensure copyright is correctly assigned

Copyright law is the bases for Open Source software licensing. Please ensure that everyfile that you can claim copyright on has a correct copyright header.

Check to see if you need a Certificate of Origin

See DeveloperCertificateOfOrigin for the exact text. A certificate of origin is required for any contribution, the only question is the form it takes.

MAINTAINER CHECKLIST

Public discussion of significant changes in API, ABI, Roadmap, etc.

If you're going to significantly change the software, those who rely on the software might be negatively affected. Its wise to discuss changes on a mailing list with a maintainer or in an issue tracker.

Do you need a mailing list and/or an issue tracker?

GENIVI provides an issue tracker, its suggested that you use that to track issues. If you can re-use an already existing mailing list to prevent mailing list proliferation, that is usually the best approach.

GENIVI has quite a few mailing lists already: https://lists.genivi.org/mailman/listinfo

Determine the compliance implications of the software

Is this software going to be part of GENIVI compliance? If so, there is a process you'll need to follow with GENIVI's SAT team. The EG lead and/or EG architect should be able to help you with that.

Prepare content for public web site

Having solid documentation is a key factor in the success of any project. In addition, blogging about new features and changes brings attention to your project so content for the GENIVI public web site is essential. Should you need it, GENIVI provides a separate documentation service: http://docs.projects.genivi.org/

CLA FAQ

Questions

Q: Where can I view the CLA?

A: Its available here as a PDF: http://www.genivi.org/sites/default/files/GENIVI_CLA.pdf

Q: I work for a company that has a CLA with GENIVI (and I am listed as one of the authorized contributors). Do I have to apply "Signed-off-by" to the patches I send?

A: Yes. If your company has signed a CLA with GENIVI you have to have a "Signed-off-by" line on your patch. In addition, you'll have to be designated as an authorized contributor by your company, otherwise GENIVI cannot accept the patch.

TOOLS IN YOUR TOOLBOX

- ➔ Public ssh key from the machine you develop on
- ➔ Subscription to (at least) genivi-projects
- ➔ git (there are numerous GUI git clients)
- ➔ Login to bugzilla, wiki
- ➔ non-html mail client (nice to have)

LINKS

GENIVI git repository
http://git.projects.genivi.org

GENIVI wiki
http://wiki.projects.genivi.org

GENIVI project pages
http://wiki.projects.genivi.org

GENIVI bugzilla
http://bugs.genivi.org

Internal OSS launch dashboard
https://collab.genivi.org/wiki/display/genivi/Open+Source+Project+Launch+Dashboard

OpenHub
https://www.openhub.net/orgs/genivi

Checklist for pushing code into the open
https://collab.genivi.org/wiki/display/genivi/Checklist+for+pushing+code+into+the+open

QUESTIONS

Q & A