STANDARDIZING THE FOSS GOVERNANCE PROCESSES
Michel Ruffin – Software coordination manager

December 2013

Alcatel·Lucent

STANDARDIZING THE FOSS GOVERNANCE PROCESSES

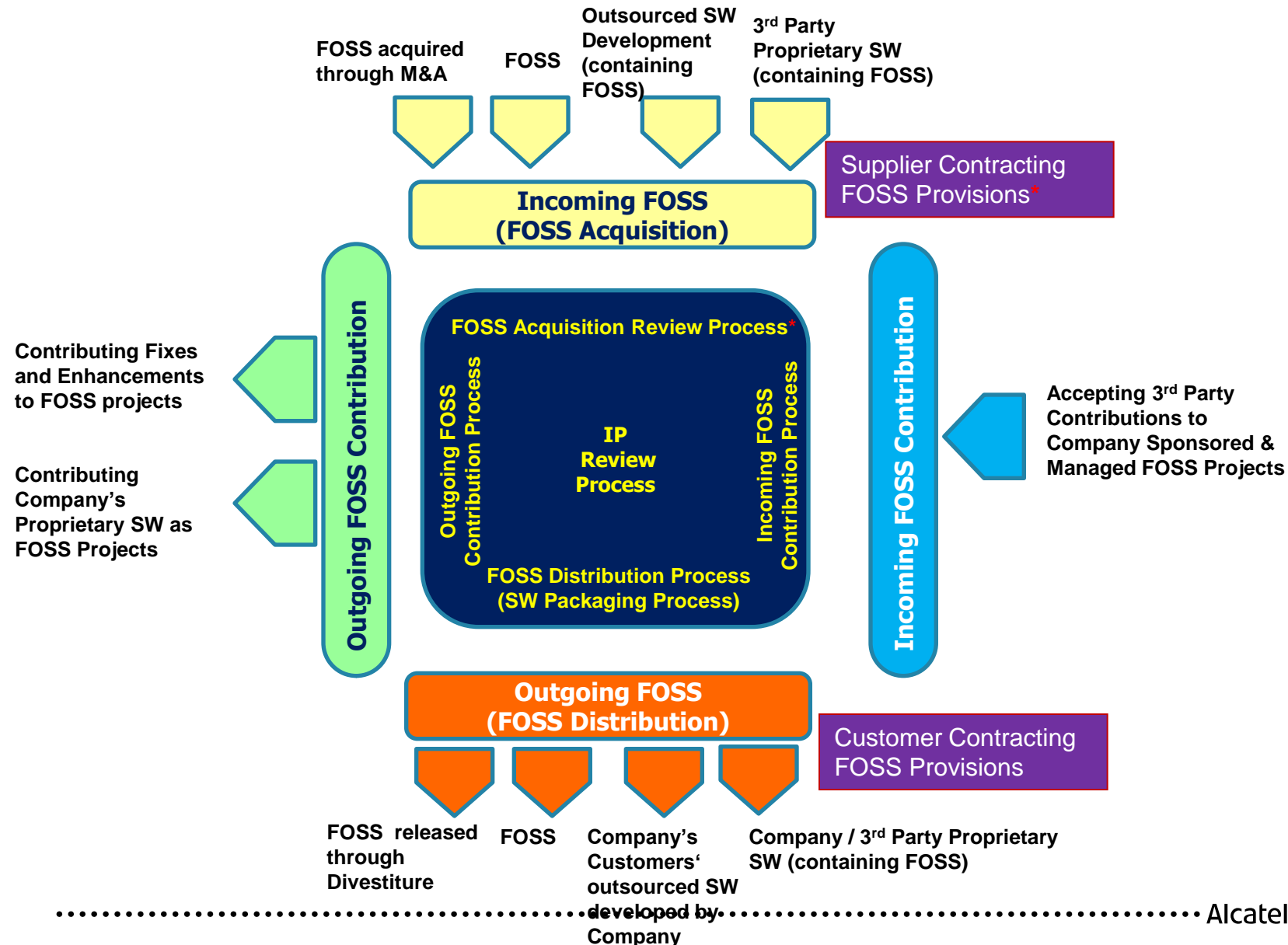## Background

- The use of Free and/or Open Source Software is increasing drastically

- Companies set up strong governance processes to respect FOSS licenses obligations

- Complexity of these processes are increasing with new technologies & business models: Maven, DRM, tivoization, BitTorrent, cloud computing (SaaS, PaaS, IaaS)

- Processes need to cope with suppliers distributing FOSS, customers receiving FOSS, mergers & acquisitions, outsourcing developments, divestitures, cloud hosting companies …

Rationale for standardizing processes

- facilitates exchange of relevant FOSS information between companies

- reduces negotiations between companies

- allows to automate part of the process

- reduces human mistakes

- raises awareness in the industry on FOSS issues (R&D, legal, procurement, quality, IT, sales people, …)

Alcatel·Lucent

SCOPE OF FOSS GOVERNANCE

FOSS acquired through M&A

FOSS

Outsourced SW Development (containing FOSS)

3rd Party Proprietary SW (containing FOSS)

Incoming FOSS (FOSS Acquisition)

Supplier Contracting FOSS Provisions*

Outgoing FOSS Contribution

Contributing Fixes and Enhancements to FOSS projects

Contributing Company's Proprietary SW as FOSS Projects

Incoming FOSS Contribution

Accepting 3rd Party Contributions to Company Sponsored & Managed FOSS Projects

FOSS Acquisition Review Process*

Outgoing FOSS Contribution Process

IP Review Process

Incoming FOSS Contribution Process

FOSS Distribution Process (SW Packaging Process)

Outgoing FOSS (FOSS Distribution)

Customer Contracting FOSS Provisions

FOSS released through Divestiture

FOSS

Company's Customers' outsourced SW developed by Company

Company / 3rd Party Proprietary SW (containing FOSS)

Alcatel·Lucent

STANDARDIZATION STATUS
FOSS IP metadata model

SPDX (Software Package Data eXchange) is the only (de facto) standard

- Codifies in XML the IP information on FOSS to exchange between organizations

SPDX still needs major improvements

- Cope with component level dependencies

  1. A SW package comes with a FOSS license
  2. The SW package can include other FOSS (component dependencies)
  3. Some files can comes with a different licenses.

- Today SPDX address issues 1 and 3 not 2 (planned to be addressed in SPDX version 2)

- Identification of FOSS need to be standardized

  - There is no unique name for FOSS
  - In Java there is the maven system (GAV: Group, Artifact, version)
  - In other languages there is no clear rule

WE NEED TO GO FURTHER IN STANDARDIZATION PROPOSAL - IDEAS

- FOSS clauses in contracts between suppliers and their customers

- But also for outsourcing, cloud hosting mergers & acquisitions and divestiture processes

- Create a database on FOSS describing Intellectual property rights (licenses dependencies, …) following some kind of Wikipedia model

  - A huge cost saving (ALU has around 200 people for feeding its FOSS DB)

  - The fields of the database must be crafted carefully to automate the generation of the document related to FOSS

- Create a training program and a certification program for educating and qualifying people

  - The major issue with FOSS is lack of knowledge

  - ALU training is one week long: 200 slides, exercises, learn to use FOSS detection tools How to package a product with FOSS for re-use by the customer

- Create a naming authority to identify uniquely

  - FOSS

  - Image, icon, Fonts (free of use)

  - Firmware

- Standardize license attributes

  - Decomposition of FOSS licenses in attributes to determine compatibilities

Alcatel·Lucent

Example: ALU Contract Conditions FOSS definition

In all ALU contracts with suppliers we negotiate FOSS clauses

- Starting with the definition of FOSS

  - Code elements subset of FOSS.

  - FOSS

    - Either OSI like

    - Or proprietary SW free of Cost

    - or shareware

    - Or public domain

  - Any SW which is not going through procurement department

  - We might not be able to standardize the legal language but the principle

Legal version

- **"Code Element"** shall mean any portion(s) of a FOSS program that are governed by license terms that differ from the license terms applicable to the FOSS program as a whole. The license terms applicable to Code Elements are sometimes referred to as "dependencies".

Free and/or Open Source Software" or "FOSS" means (i) software provided to Licensor royalty-free in source code form, under a license including, but not limited to, one approved by the Open Source Initiative (OSI http://www.opensource.org/) or (ii) proprietary software provided to Licensor royalty-free in binary code form, under an end user license agreement that is accepted without a signature, or (iii) shareware provided to Licensor free of initial charge, such as on a trial basis, but where a fee may become due once the user decides to use the software beyond the trial period, or (iv) public domain software

Alcatel·Lucent

Example: ALU FOSS clauses (1/2)

Clause 1

- The list of FOSS must be provided by the supplier: FOSS name, version, license name (and version), Nature of the FOSS (standalone SW, library, interpreter, script…)

- The supplier should try to negotiate the condition with its suppliers

- The information should be provided in excel of SPDX format

Clause 2

- Supplier respects the FOSS licenses

- Licenses are compatible with each other.

- In case of contradiction between the FOSS licenses and the contract, ALU can can apply the FOSS license, while keeping the benefit of any terms in the contract that are "better" than corresponding terms in the FOSS license. Examples include indemnity and warranty provisions, which would extend to any FOSS included in supplier's software.

Alcatel·Lucent

Example: ALU FOSS clauses (2/2)

<u>Clause 3</u>: requires the supplier to provide source code if need be, if the FOSS license requires source code availability

<u>Clause 4</u>: Supplier must supply to us the packaging information

- What we have to put in our distribution

- What we put in our documentation

<u>Clause 5</u> : We want from FOSS distributors to have FOSS under FOSS license terms if need be to avoid supplier lock-in.

- No sub-licensing

- We want to be able to provide the source code to another company for getting support without bad terms and conditions

All these clauses are quite simple but generate a lot of discussions between lawyers

Standardizing these clauses (or their principle) would save a lot of manpower

Why the OMG?

• The Linux foundation/FOSSBazaar/SPDX people will continue the development of the SPDX standard but seems not yet interested in addressing other FOSS standard activities

• The Free Software Foundation European (FSFE) network seems not to be interested in this standard topics

• The OMG (Object Management Group)

- Is a de jure standard.

- Has experience in certification programs

- Is well connected all around the world

The challenge

Since it is all about intellectual property issues the legal aspects on these action will need careful legal consideration