

# Legal issues in Open Source Community and Project Management

# Main legal decisions

- Legal structure
- IPR management
- Branding (trademark)

# Legal structure

- Formal legal entity
  - Fundación
  - Asociación
  - Consorcio
- Informal legal structure
  - Contractual relationship (interadministration?)
  - Nothing!

# Terms of Governance

- Formal structure
  - Foundational document (statutes, by laws, etc.)
  - Internal regulation
- Any structure
  - Governance rules
  - Contribution policy
  - Trademark policy
- Decidim:
  - Social Contract (who does this apply to)
  - Guifi.net: comuns de la XOLN

# Tisano (health application)



public officials from state level and local- local

knowledge of the business, background, practices and processes, use case, user testing and acceptance of the code, customer discovery  
1-2 / week (tel conf)

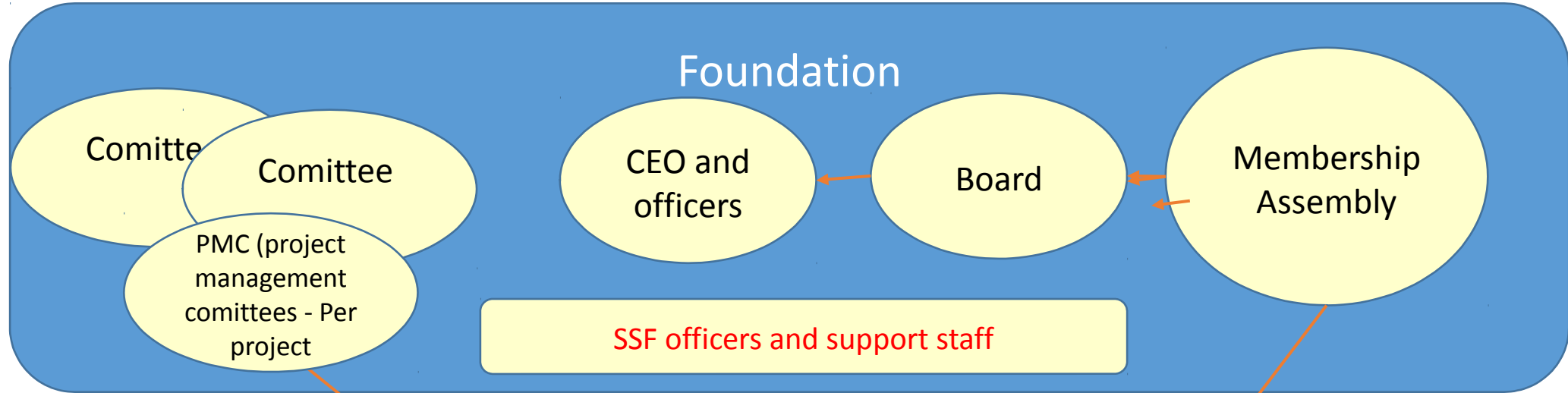
Corporate Contributor License Agreement

broader issues, market impacts, "big picture", reviews project progress, burn rate .  
insight or guidance on other events  
1 / quarter

Membership Agreement + Statements of Work

*Business applications require a non-technical resource to drive the development.  
Need to harvest the knowledge of subject matter experts, to create and support the community and have a third party to bring the SME team together becomes critical*

# SAHANA



operational, legal and procedural oversight on Sahana releases  
**meritocracy**

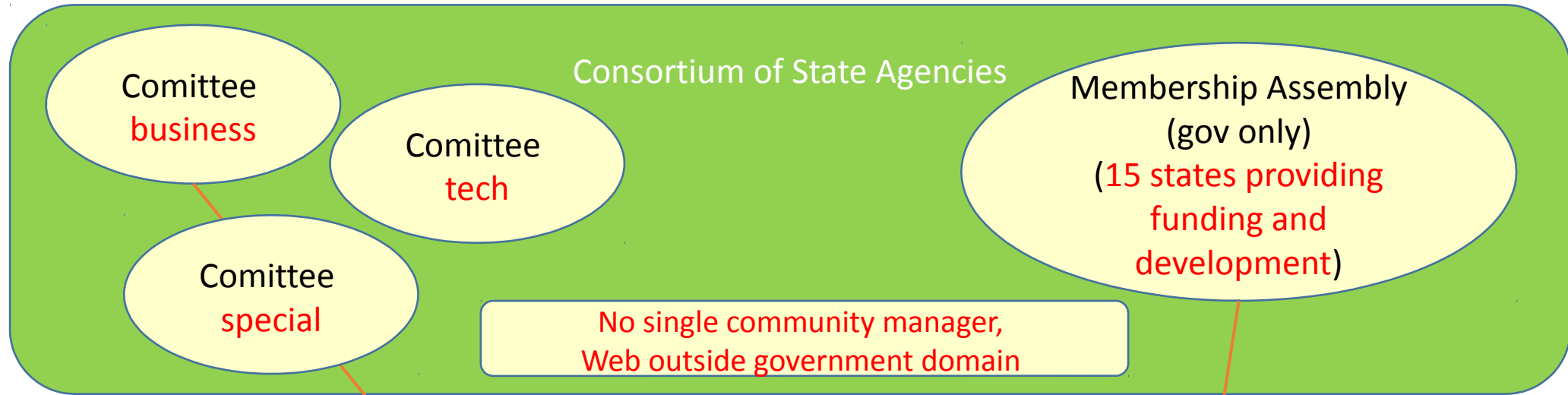
Contributor License Agreement

Overall official planning  
Receives donations

bylaws and Governance structures of the projects

*meritocracy highlighted in the community operating model encourages and rewards volunteer contributors through being invited to be a member of a Project Management Committee*

# NCOMS (offender information)



collaborative technological  
goal-setting, standards  
creation, architecture  
frameworks discussions,  
and broad term user  
community issues

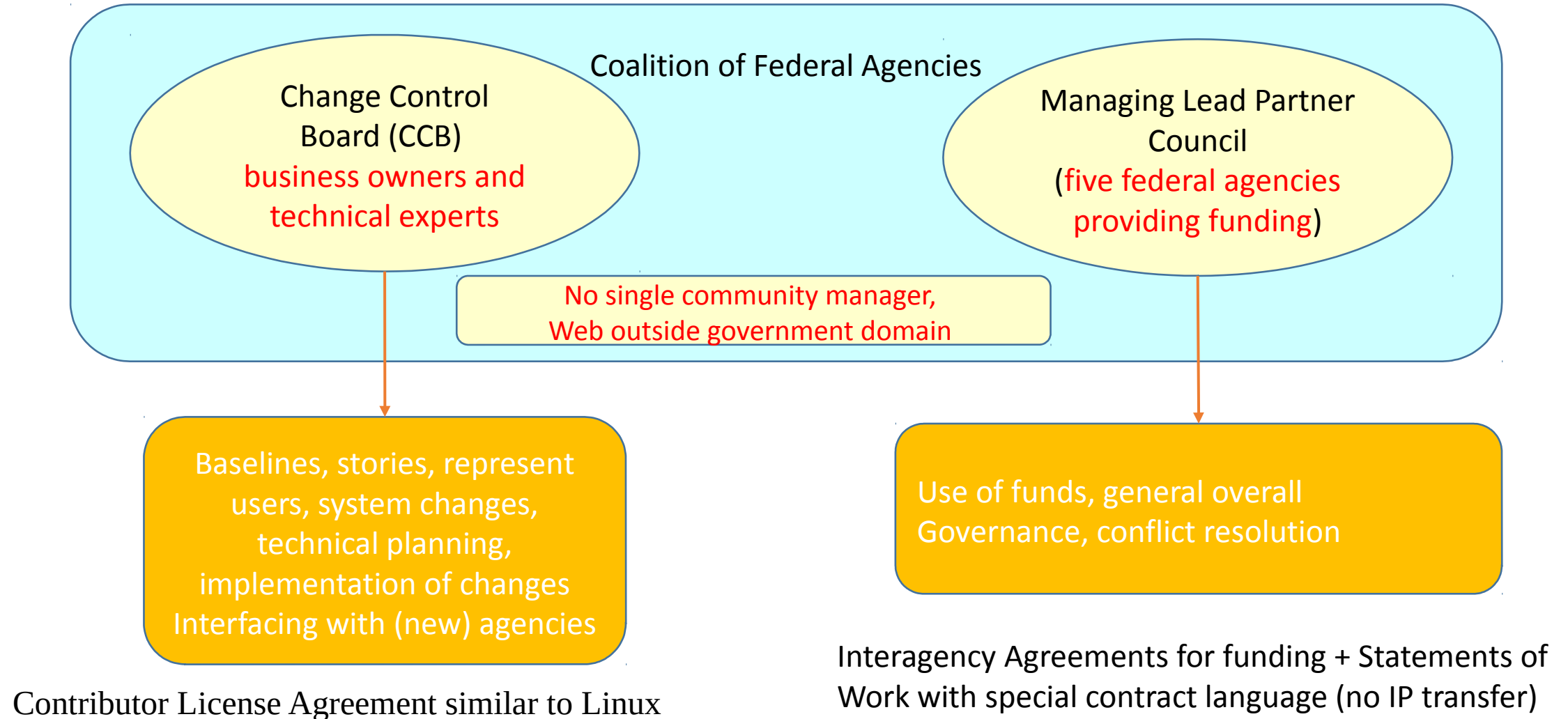
Contributor License Agreement similar to Linux

leadership focused, consider leveraging  
budgets cycles, legislative trends,  
developing trust, and re-emphasizing  
the importance of communication and  
collaboration

Government Agreement re IP ownership +  
Consortium bylaws

*Need to set up formal structure (NCOMS lacked one for 6 years). Participation of private developer companies*

# CONNECT (health information application)



*Need to involve actual users and contributors in management (currently very federal agency oriented)*

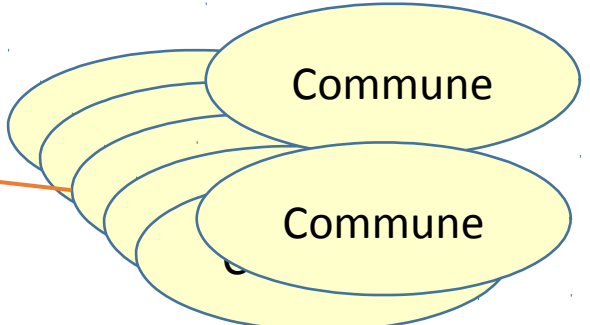
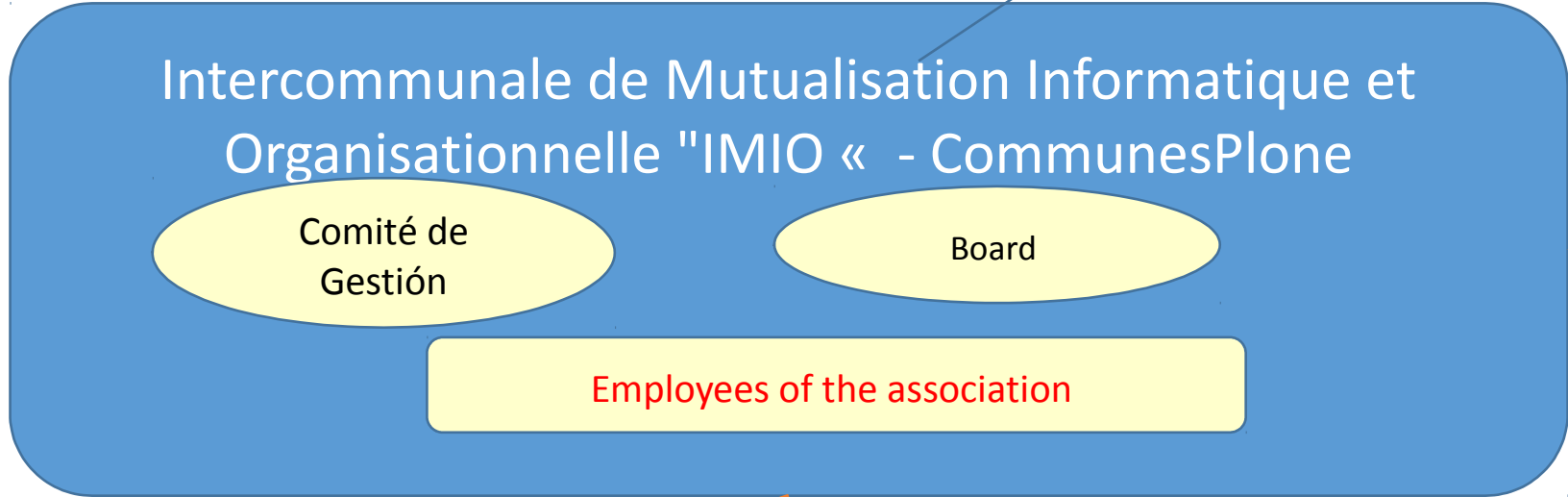
*Create foundation ? Balance public funding and private inputs*

*key role bringing in someone with deep experience in work on large, successful open source projects as a consultant*



PloneGov

association intercommunale  
Société coopérative à responsabilité limitée



bylaws and Governance structures of the projects



GPL

*No legal structure until 2012*

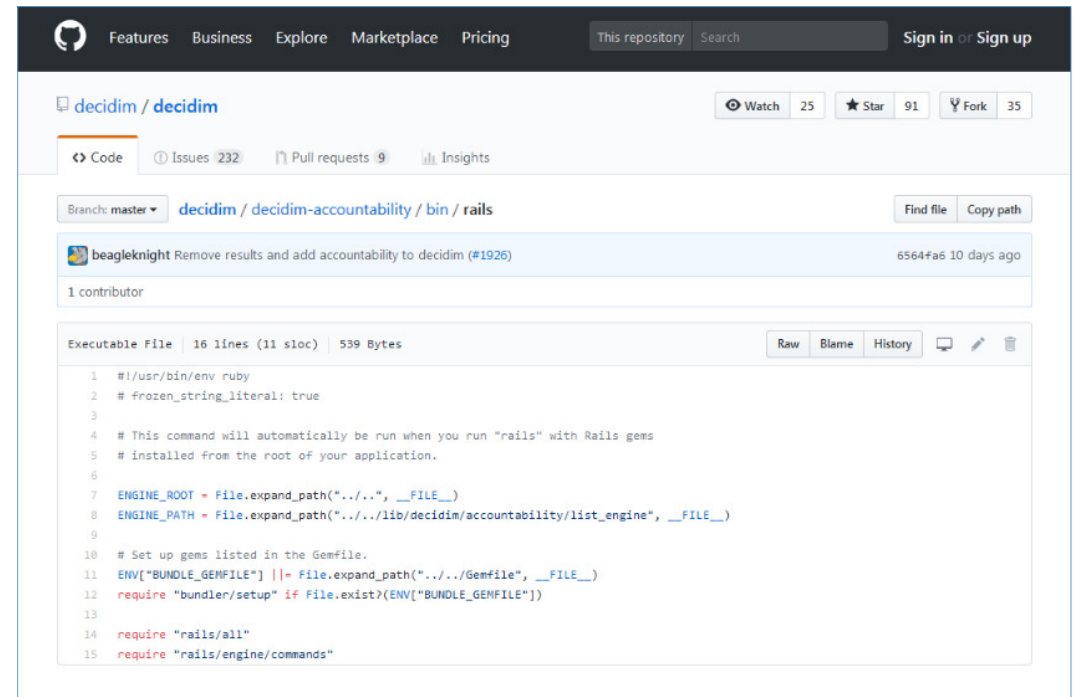
# Branding - trademark

- Registration process (cost)
  - Per territory, products and services
- Exclusive use of the mark for the owner
  - Use in commerce
- Licensed use
  - For community users
  - For comercial purposes
- Questions:
  - What trademark, what territory, what logo?
  - Who is the owner
  - What licensed rights should be granted to the community



# IPR management

- Copyrights in the code
  - Initial ownership of rights
  - Contribution policy
    - Assignment of rights (CLA)
    - Contributor License (CLA)
    - Project license (DCO)
- Copyrights in materials, web, texts, designs,
  - Idem
- What license? Or “licensing” (various licenses?)
- IPR Compliance – verifying license terms and compatibility
- **Importance: owner decides on license, license change, license alternatives, legal defense... and assumes responsibility for infringements**



The screenshot shows a GitHub repository page for 'decidim / decidim'. The commit history shows a commit by 'beagleknight' titled 'Remove results and add accountability to decidim (#1926)' made 10 days ago. The diff view shows changes to the file 'rails' in the 'bin' directory. The code is as follows:

```
1 #!/usr/bin/env ruby
2 # frozen_string_literal: true
3
4 # This command will automatically be run when you run "rails" with Rails gems
5 # installed from the root of your application.
6
7 ENGINE_ROOT = File.expand_path("../..", __FILE__)
8 ENGINE_PATH = File.expand_path("../lib/decidim/accountability/list_engine", __FILE__)
9
10 # Set up gems listed in the Gemfile.
11 ENV["BUNDLE_GEMFILE"] ||= File.expand_path("../Gemfile", __FILE__)
12 require "bundler/setup" if File.exist?(ENV["BUNDLE_GEMFILE"])
13
14 require "rails/all"
15 require "rails/engine/commands"
```