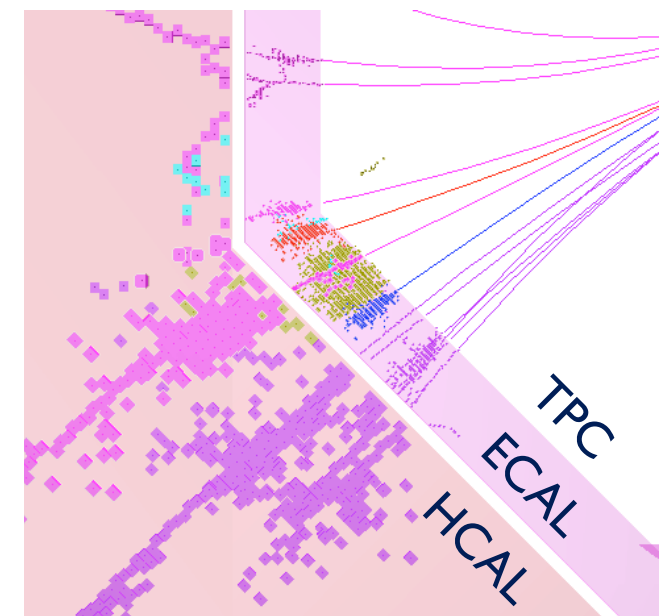




Pandora LArSoft Integration

A. S.T. Blake, J. S. Marshall, M.A. Thomson
16 February 2016





Pandora LArSoft Integration

REMINDER:

- Pandora brings a multi-algorithm approach to LAr TPC pattern recognition. The Pandora SDK (EPJC.75.439) was engineered to provide an environment in which:
 1. It is easy for users to provide the building-blocks that define a pattern recognition problem.
 2. Logic required to solve pattern recognition problems is cleanly implemented in algorithms.
 3. Operations to access or modify building-blocks, or to create new structures, are requested by algorithms and performed by the Pandora framework.

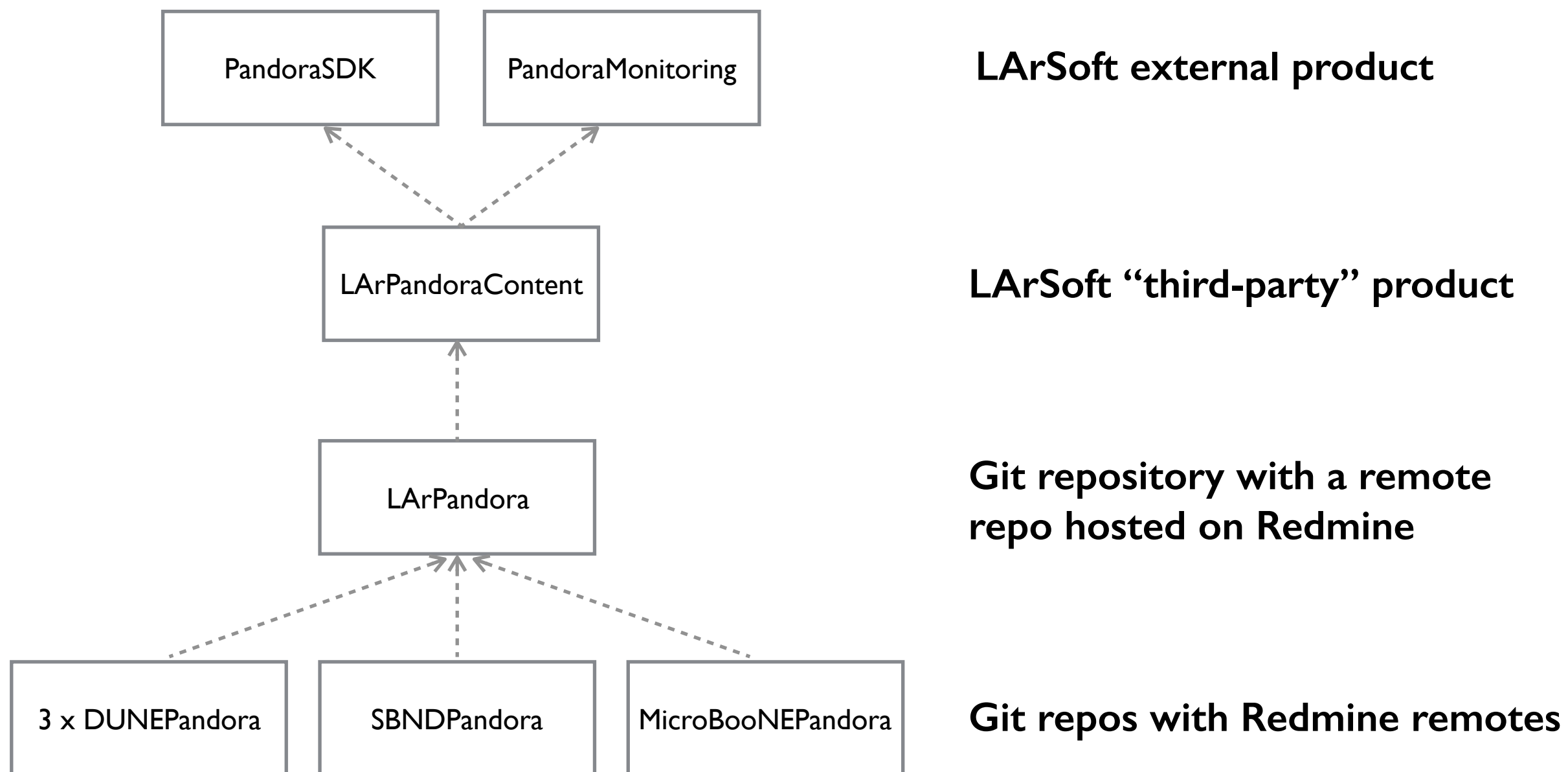
TODAY:

- Review of the current Pandora footprint in LArSoft.
- Proposal to ease distribution of Pandora algorithm changes to LArSoft users.
- Description of recent changes to the LArPandora “translation” package.



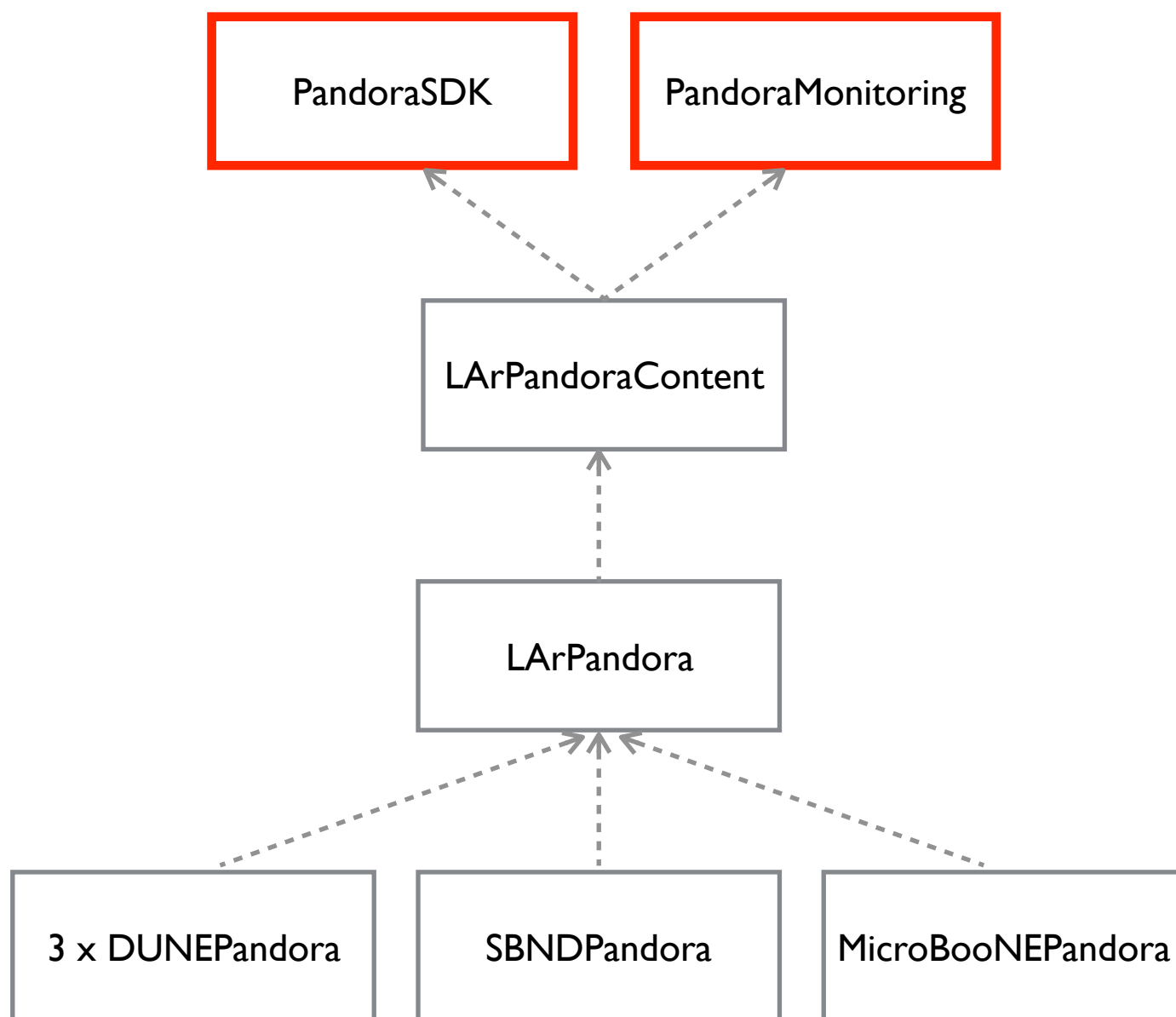
Pandora in LArSoft

Simple cartoon showing current packages and an indicative hierarchy:



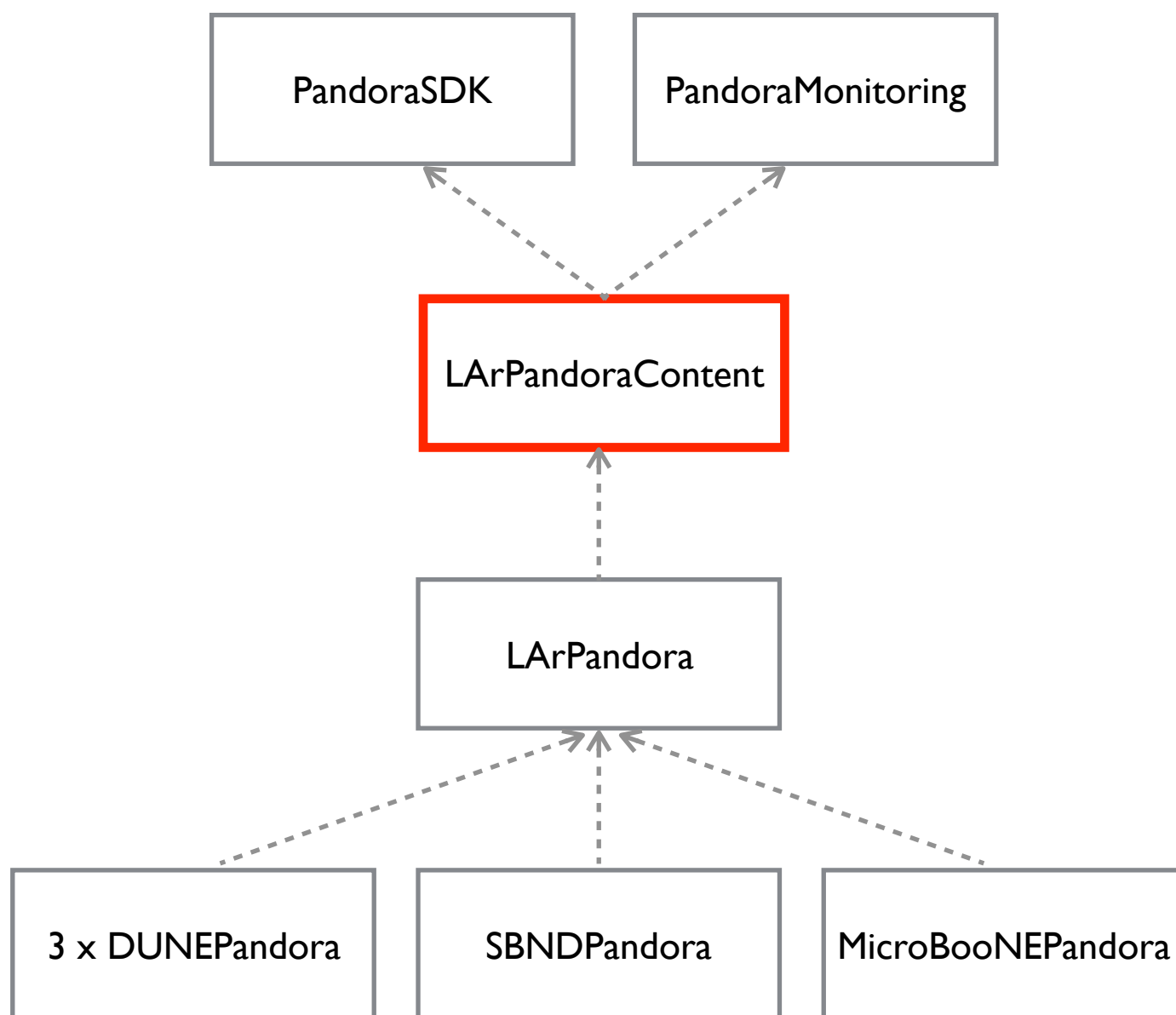


Pandora SDK and Monitoring



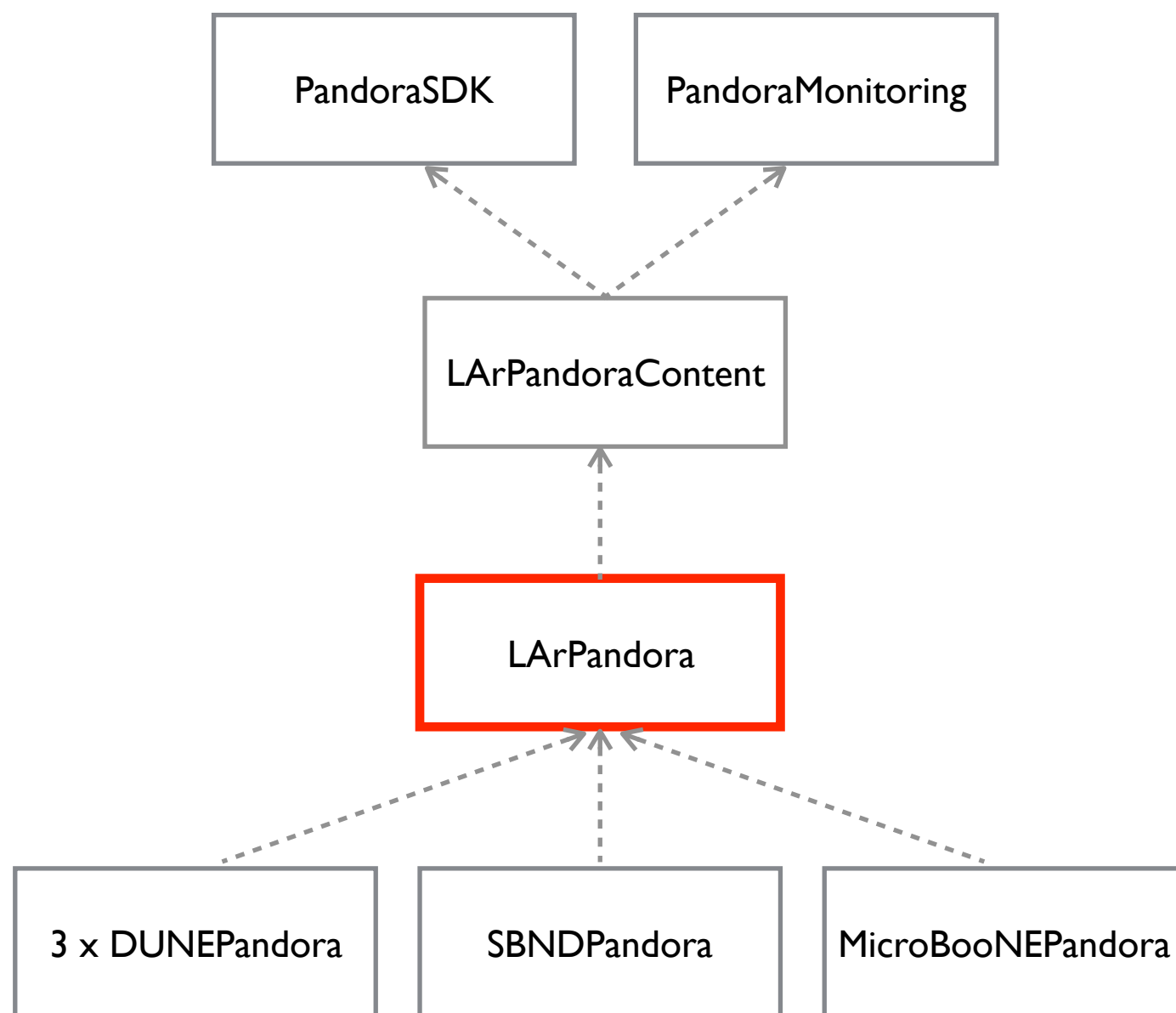
LArSoft external product

- Product “pandora” consists of two libraries and their associated APIs.
- Common to LC and LHC use-cases.
- PandoraMonitoring depends on ROOT.
- PandoraMonitoring not fully-functional in LArSoft; cannot create a TEveManager.
- **Git remote repositories at github.com/PandoraPFA**



LArSoft “third-party” product

- Contains Pandora LAr TPC pattern recognition algorithms and tools.
- Depends on Pandora SDK and (optionally) Pandora Monitoring.
- Under active development, but only formal releases available in LArSoft.
- Significant overheads whenever we want to distribute updates. Little flexibility.
- Where our “real work” is done and would like to attract new developers.
- **Git remote repository at github.com/PandoraPFA**

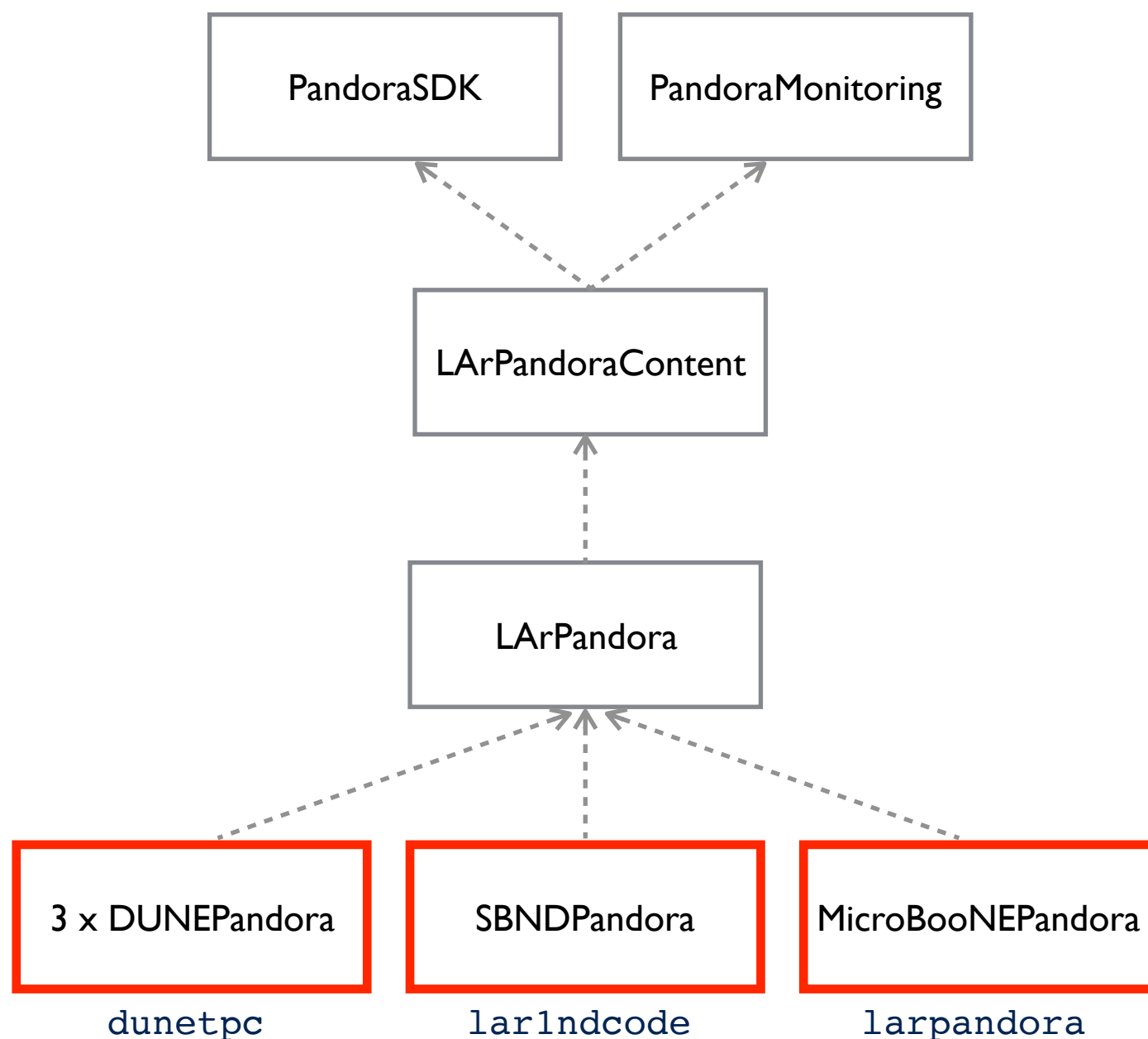


Git repo with Redmine remote

- Interface and common implementation classes for Pandora modules in LArSoft.
- Translation package, depends on both Pandora and LArSoft packages.
- Turn LArSoft Hits and MCParticles into Pandora CaloHits and MCParticles.
- Turn Pandora Particles into LArSoft PFParticles, with correct associations.
- **Aim: do very little, other than simple, unit-tested translation mechanics.**



MicroBooNEPandora, etc.

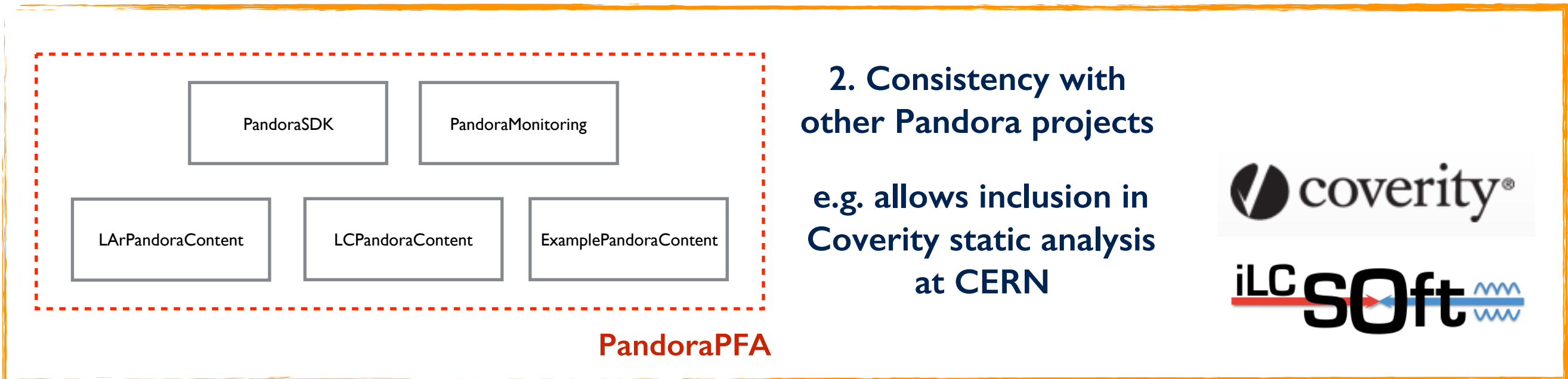
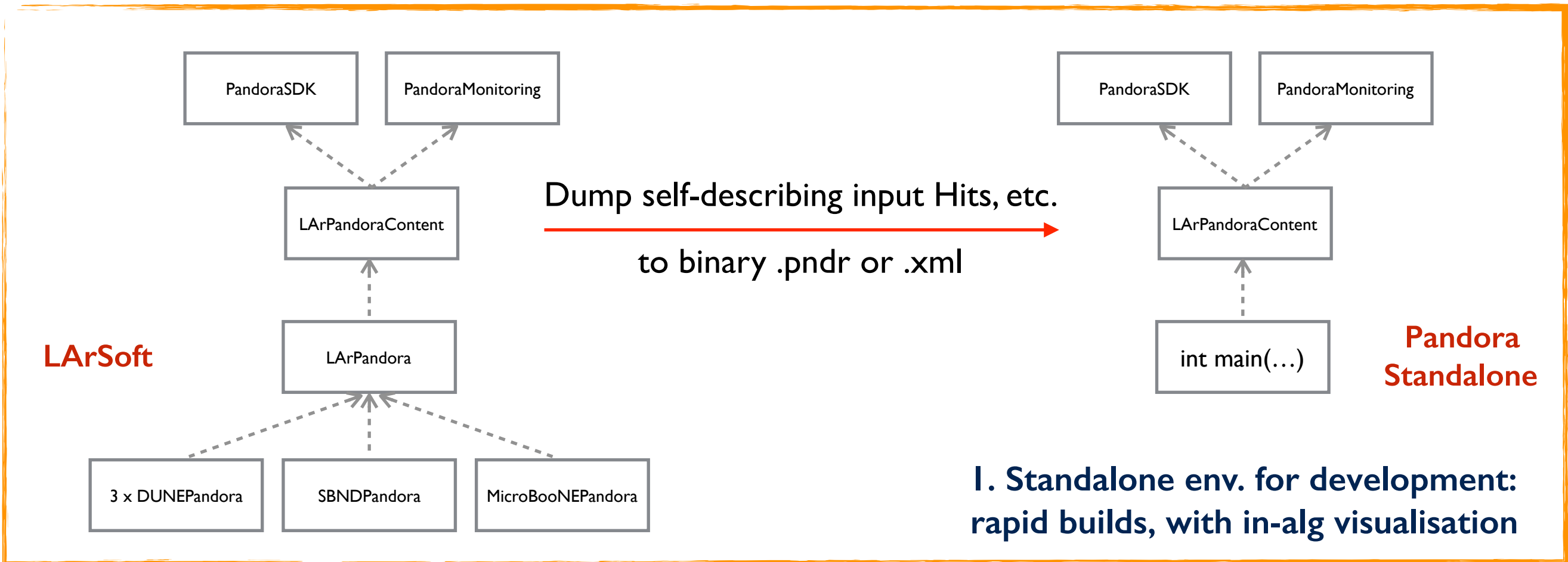


Git repos with Redmine remotes

- Know about specific wire-pitches, wire-angles, etc.
- Create Pandora instances, register coordinate transformation plugins.
- Not really needed, if suitable geometry abstraction available to LArPandora.
- Support concept of a “drift volume”.

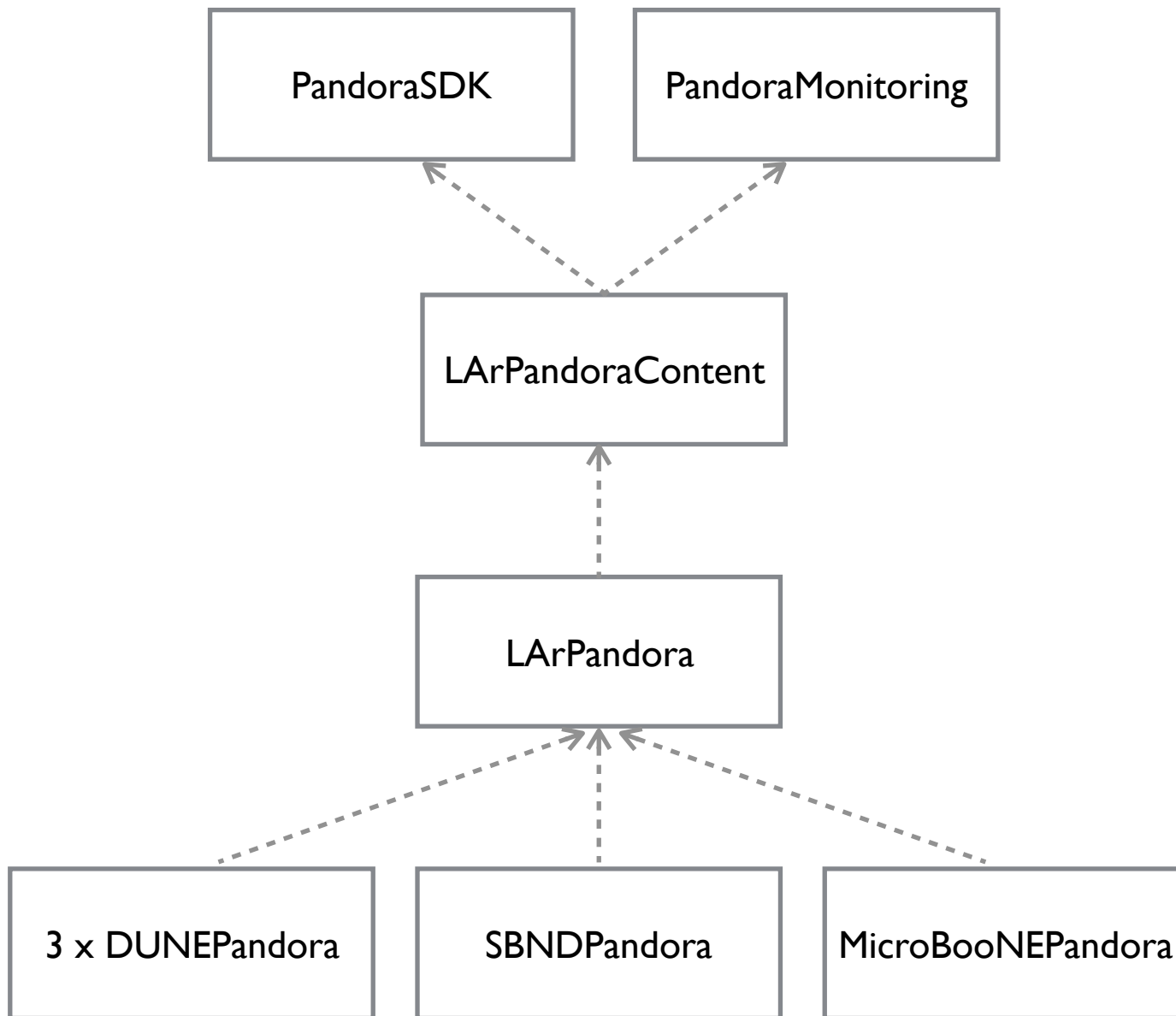


External Constraints





Proposal



LArSoft external product

~~LArSoft “third-party” product~~
Git repo with Redmine remote

Git repo with Redmine remote

Git repos with Redmine remotes



Proposal

Redmine

Remote repo designated for larsoft releases and a go-to for all LArSoft developers

GitHub



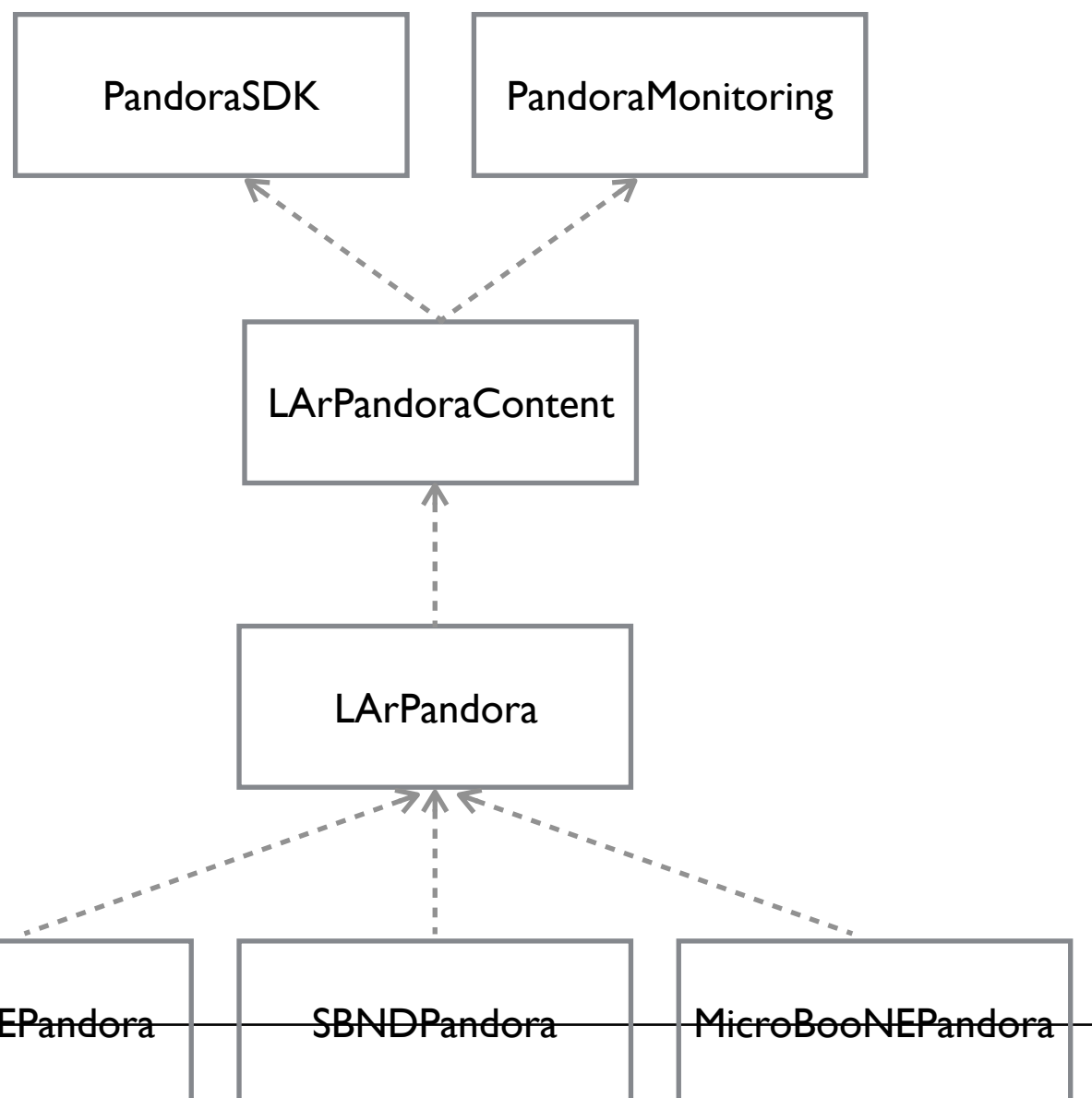
```
git remote add redmine <url>
git fetch redmine
git checkout -b mybranch
git merge redmine/develop
git push origin mybranch:myremotebranch
```

E.g. More (and better) examples online. Redmine repo will probably track our github repo to start with.

- **Use git functionality to deal with multiple remotes.** Pandora pattern recognition algorithms become a proper part of LArSoft. Existence of other remote repos is transparent to LArSoft.
- New Pandora algorithm functionality quickly available to LArSoft users e.g. via feature branches on Redmine remote. Still have formal dialogue for pull requests to master branch.
- One tension is CMake build configuration, which should support both existing/external Pandora approach and LArSoft build tools. Suggest simple "IF (LARSOFT) ..." in CMakeLists.txt



Minor Additions?



Update periodically, to pick-up e.g. new LC/LHC additions, even if no LAr TPC impact?

~~LArSoft “third-party” product~~
Git repo with Redmine remote

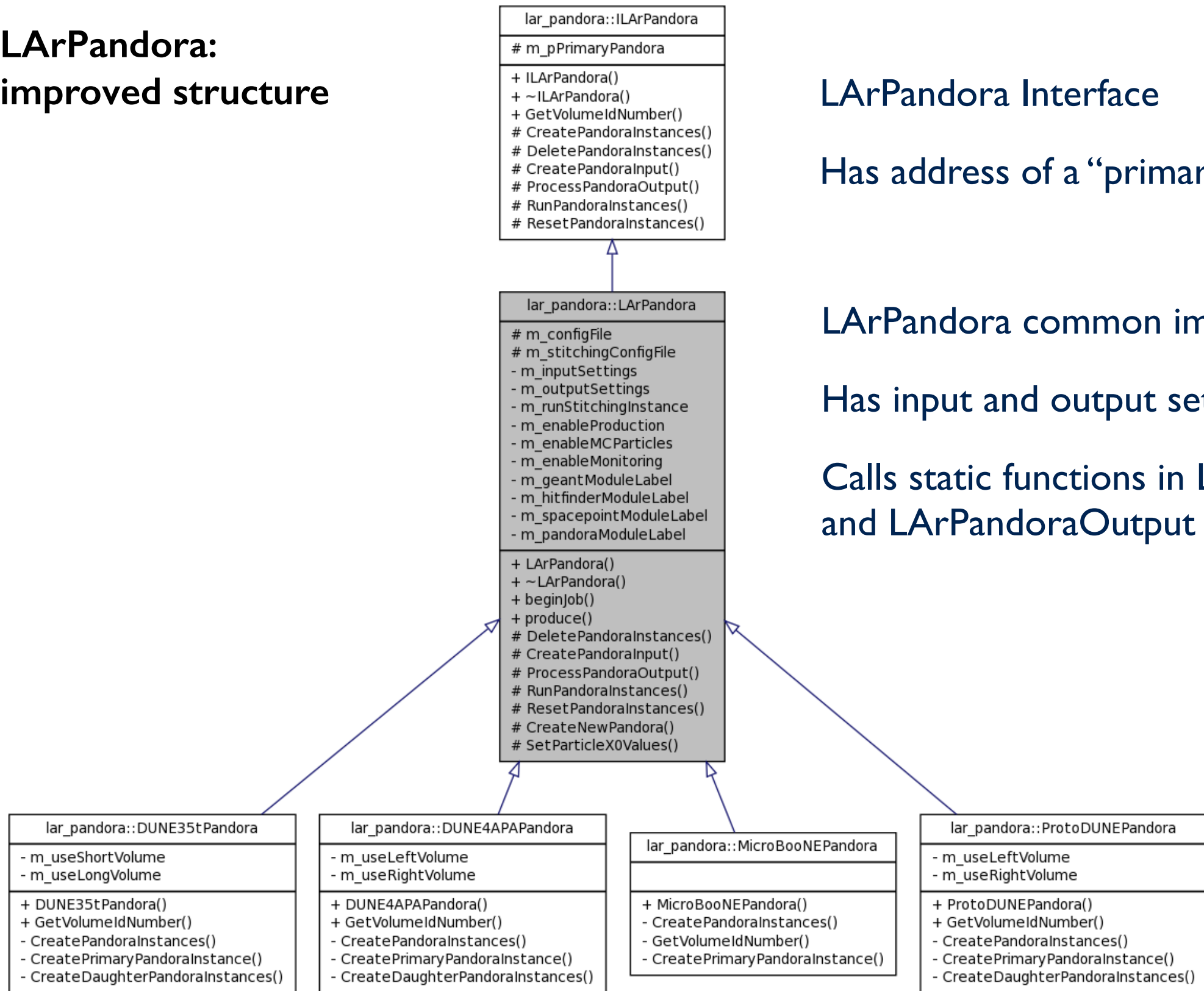
Git repo with Redmine remote

Fold functionality into LArPandora, if/when geometry abstraction allows?



Recent Changes: LArPandora

LArPandora: improved structure



LArPandora Interface

Has address of a “primary” Pandora instance

LArPandora common implementation

Has input and output settings instances

Calls static functions in LArPandoraInput and LArPandoraOutput

Create Pandora instances

Define drift volumes



Recent Changes: MultiPandoraApi

LArPandora producer modules use MultiPandoraApi to handle multiple Pandora instances.

One “daughter” instance to reconstruct each volume, plus primary instance to stitch-together particles crossing between volumes.

```
/**
 * @brief MultiPandoraApi class
 */
class MultiPandoraApi
{
public:
    /**
     * @brief Declare a new primary pandora instance
     *
     * @param pPrimaryPandora the address of the primary pandora instance
     */
    static void AddPrimaryPandoraInstance(const pandora::Pandora *const pPrimaryPandora);

    /**
     * @brief Add a pandora daughter instance, associated to a primary pandora instance
     *
     * @param pPrimaryPandora the address of the primary pandora instance
     * @param pDaughterPandora the address of the daughter pandora instance
     */
    static void AddDaughterPandoraInstance(const pandora::Pandora *const pPrimaryPandora, const pandora::Pandora *const pDaughterPandora);

    /**
     * @brief Get the address of the daughter pandora instance associated with a given primary pandora instance and volume id number
     *
     * @param pPrimaryPandora the address of the primary pandora instance
     * @param idNumber the volume identifier number
     *
     * @return the address of the daughter pandora instance
     */
    static const pandora::Pandora *GetDaughterPandoraInstance(const pandora::Pandora *const pPrimaryPandora, const int idNumber);

    /**
     * @brief Delete all pandora instances associated with (and including) a specified primary pandora instance
     *
     * @param pPrimaryPandora the address of the primary pandora instance
     */
    static void DeletePandoraInstances(const pandora::Pandora *const pPrimaryPandora);

    /* ... SNIP ... */
};
```



Thanks for your attention!