System:

- Create new namelists: How to create new namelists to be used by HARMONIE
- Harmonie testbed: The HARMONIE test environment used to test typical configurations on a small domain
- Phasing instructions
  - Access to Météo-France servers
  - Mitraillette
- Profiling with DrHook: ECMWF's profiling and traceback tool included in the code
- HARMONIE RAPS benchmark
- Meteo France/ECMWF 2011: Coding standards for ARPEGE-IFS

Code browsers

- Code browsers (ftagshtml), 36h1.3, 37h1.1, 38h1.beta.1
- Doxygen code browser (36h1.4)
- Trac code browser

HARMONIE Reference materials

- **Excellent ALADIN/Arome documentation**
- IFS documentation
- SURFEX documentation
- Meteo France/ECMWF 2011: Coding standards for ARPEGE-IFS
- Guidard Vincent and C. Fischer, ALADIN 3D-Var Theory and Practice
- Kert'esz, S., 2001: A short overview of ODB
- Kert'esz, S., 2002: Using ODB in ALADIN. Internal note, 21pp, available on " http://www.cnrm.meteo.fr/gmapdoc/" (topics "ODB")
- M. Lindskog, 2010: Study with extended extension zone, manuscript for HIRLAM Newsletter 2010
- R. Roger, The data flow in HARMONIE data assimilation
- S. Saarinen: Initial Optimisation of AROME forecasts on FMI's Cray XT5m, Presentation at ECMWF, Nov 20 2009
- Yann Seity's unofficial documentation about Surfex surface drag options. Note the options described are those featured in CY36 and CY38, and some of the options for CY38 need back porting to CY37.
- Trojakova Alena, CANARI optimum interpolation with emphasis on technical aspects
- Yessad K.: Aug 2007: LIBRARY ARCHITECTURE AND HISTORY OF THE TECHNICAL ASPECTS IN ARPEGE/IFS, ALADIN AND AROME IN THE CYCLE 32T2 OF ARPEGE/IFS
- Yessad, K., 2007: Basics about ARPEGE/IFS, ALADIN and AROME in the cycle 32T2 of ARPEGE/IFS

- Vignes, O., 2011: Short presentation of LSMIXBC option
- Bengtsson, L., 2014: CA scheme description

HIRLAM system revision control ¶

Starting from 2006, HIRLAM-A chooses Subversion as its revision control system. The first Hirlam release with Subversion feature is Hirlam-7.0beta1, release in March 2006.

HIRLAM system versioning ¶

HIRLAM repository naming convention ¶

In the past few years, Hirlam tags an official system release about once a year. The official Hirlam release is named in form of Hirlam-l.m(.n), where

- "l" refers to the current HIRLAM programme, l=6 for Hirlam-6 project, l=7 for Hirlam-A.
- "m" is the serial number of the official release during the programme period, (for Hirlam-A, the official Hirlam releases consists of Hirlam-7.0, Hirlam-7.1, Hirlam-7.2, Hirlam-7.3, Hirlam-7.4...)
- "n" is not assigned at initial official release (and equivalent to zero). When a bugfix version or other technical update is tagged, "n" is assigned serial number 1, 2, 3, 4.
  - e.g., for Hirlam-7.2 released in Sept 2008, a follow up release Hirlam-7.2.1 is tagged in April 2009 to update the default computer platform from ecgate/HPCE to ecgate/C1A, together with a few bug corrections.

Official HIRLAM releases ¶

Hirlam system releases are feature based, each of them associated with certain major meteorological or technical features, which are normally declared in the release notes. An officially released Hirlam system shall satisfy some minimum requirements,

- That the release comes with new meteorological or technical features which overall outperform the previous release
- That the above required validatation has been done at the reference computer platform, i.e, ECMWF HPC
- That the release candidate has been in pre-operational status at the FMI for positive evaludation prior to tagging, with participation of duty forecasters and scientific developers

The decision about the official release of the new Hirlam system is taken by the HIRLAM management group, with close consultation with FMI. The latter commits to use the official reference release as its main production suite.

HIRLAM repository with Subversion revision control ¶

Hirlam system repository is maintained by the system group (system@hirlam.org) lead by the HIRLAM-A project leader on system

and applications. The repository is maintained on hirlam.org using Subversion tool. The collection of system codes (including that of source code, scripts, build tools and other utilities) is organised in form of subversion trunk, branches and tags.

- Trunk
  - trunk is the main development code set as candidate for the next official system. e.g.,
    - Before the launch of Hirlam-7.3, Hirlam trunk is always the latest update for 7.3.
    - After tagging of 7.3, trunk becomes candidate of 7.4
  - trunk is updated continuously, merging code updates submitted by various development teams. The candidate codes for merges can be delivered via email, tarball or repository (branches).
  - Warnings!! Trunk is a moving system. Make sure that you do not base your experiment on a trunk sandbox which itself changes.
- Branches
  - Technically branches is similar to trunk, but for various development purposes
    - national branch, in which usual configuration deviation from 'reference' features in the trunk are maintained.
    - special feature branch in research/development stage (newsnow, cis)
    - stable branch (such as 7.2 stable branch which is set up after tagging of 7.2. The stable branch does not take in new features, but to collect essential bug fixes for the next tagging)
  - Using Subversion tools for merge/update, the features among branches and trunks can be merged to each other conveniently.
- Tags
  - Tagged codes are anapshots of trunks or branches. Technically the tagged codes are no different from those trunk or branches, but for convention that no longer get further updated.
  - In Hirlam, various tagging prior to official releases are made to provide user communities with a 'frozen' code set, even though the code has not necessarily been fully validated. These codes are often labeled as alpha, beta, rc.
    - Alpha release (e.g., 7.3alpha): a snapshot of trunk which is unmature both technically and meteorologically
    - Beta release (e.g., 7.3beta1): a snapshot of trunk which is deemed technically mature for evaluation and meteorological validation. On the other hand,

there could still be possibility for more features to add

- Release candidate(e.g.,7.2rc1): pre-release tagging for final evaluation
  - Official release (e.g., 7.2): mature for operational use

Post-release tagging (e.g., 7.2.1): bug fixes, platform change hpce-c1a;...