

Release coordination documentation

- Preparation
 - Preparing this document itself
 - Preparing JIRA tickets
 - Declaration of Intentions
 - Update your checkouts
 - Environment variables
 - Compara servers
 - Guidelines for the deletion of Databases
 - Configuration file
- Schema preparation
 - Update table.sql and create patch files
 - Check the patch files
 - How to patch a database to the latest schema
- Master database
 - NCBI taxonomy data
 - Add new entries to compara master database
 - Add method_link_species_set entries to compara master database
 - Add new species to phylogenetic tree
 - Final edits to compara master database
- Production
 - Main site
 - GRCh37 site [edit]
 - Connection details
 - Case 1: no data updates
 - Case 2: data updates
 - Past updates:
- End of production window
 - Create Release Database
 - Merge DNA data
 - Merge GeneTrees+Families+NCTrees+PatchProjectionsAsHomologies
 - Final database checks
 - Run the healthchecks
 - Test web server
 - Final handover of databases [edit]
 - Handover
 - Final bits
- Post-handover
 - Update documentation and diagrams [edit]
 - Patch the test databases
 - Branch the code
 - Specialize the branch
 - Test the sites
 - Data dumps

Preparation

Preparing this document itself

This document is usually inherited from the previous release cycle and tends to have all the check-boxes ticked. The fastest way to untick them all to start afresh is to go to "Edit...", then switch over to the XML view (a button on the top right with <> on it), and then perform a mass-replace of `<ac:task-status>complete</ac:task-status>` by `<ac:task-status>incomplete</ac:task-status>` .

Preparing JIRA tickets

As of release 89, JIRA ticket creation for the release process is performed automatically using a script. The script assumes that the relco is running it, but you can use the `-relco` option to override.

Automatically create JIRA tickets

```
cd $ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/jira_tickets
perl create_compara_release_JIRA_tickets.pl -tickets
Jira_recurrent_tickets.txt -c AutomaticJiraTickets.conf -release XX
```

Declaration of Intentions

- ☒ Once the release coordinator has sent out the email for the declaration of intentions, set up a web page with intentions in the Confluence wiki system to allow easy tracking of the progress. Release plans
- ☒ Ask compara team members of their intentions
Compara has one extra day to declare their intentions because of the need to know what the genebuilders and associated teams (eg wormbase, ensembl genomes) will declare
- ☒ Submit the declaration of intentions on the <http://admin.ensembl.org/index.html> website
 - ✓ [Click here to expand...](#)
 - Give a short and meaningful title
 - Leave "*Species affected*" to "*All species*"
 - Describe the change in "*Content*"
 - "*Status*" is probably "*Declared*" at this stage, but it may happen that you can already mark it as "*Handed over*"
 - Change the "*Site Type*" to "*GRCh37*" if needed
 - The default values of the remaining fields are fine
- ☒ Discuss with the team which declarations are worth being put on the front page (3 at most). For those, leave "*Headline position on the homepage*" to "Not a headline" but fill in a correct *category* ("API/Schema changes", "New alignments", etc)

Update your checkouts

Ensure you have up-to-date git checkouts of at least the following repositories, pointing at master branch:

- ☒ ensembl-compara
- ☒ ensembl
- ☒ ensembl-hive
- ☒ ensembl-analysis
- ☒ ensj-healthcheck
- ☒ ensembl-production
- ☒ ensembl-orm
- ☒ ensembl-taxonomy
- ☒ public-plugins
- ☒ ensembl-test

Environment variables

- ☒ Define \$ENSEMBL_CVS_ROOT_DIR
This is necessary to run the Hive and is used by many scripts/files in this document. Make sure this is defined in your terminal
- ☒ Define \$ENSADMIN_PSW
The password for the mysql 'ensadmin' user also needed for many scripts
- ☒ Define \$COMPARA_REG variable to simplify connecting to databases via registry

```
export COMPARA_REG="-reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -reg_type compara -reg_alias"
```

- ☒ Define \$CURR_ENSEMBL_RELEASE variable to simplify naming of databases

```
export CURR_ENSEMBL_RELEASE=`perl -mBio::Ensembl::ApiVersion -e 'print
Bio::Ensembl::ApiVersion::software_version()."\n"'`

# make sure that you got the right value (your ensembl checkout has
to be up-to-date)
echo $CURR_ENSEMBL_RELEASE

# it is also very handy to have the previous release number in a
variable:
export PREV_ENSEMBL_RELEASE=`expr $CURR_ENSEMBL_RELEASE - 1`
```

- ☒ source the RH7 env

```
source /nfs/software/ensembl/RHEL7/envs/basic.sh
```

Compara servers

Check out the current space on the compara servers and delete the last but one release. Leave the previous release for healthchecks. Check with the other compara team members before deleting.

- ☒ Check space here

- Check space on:

- ☒ <http://ens-prod-1.ebi.ac.uk:5005/status/mysql-ens-compara-prod-1>
- ☒ <http://ens-prod-1.ebi.ac.uk:5005/status/mysql-ens-compara-prod-2>
- ☒ <http://ens-prod-1.ebi.ac.uk:5005/status/mysql-ens-compara-prod-3>

- ☒ Ask compara team members to tidy up any unwanted databases (run the command below for all compara servers) and inform them of the intention to delete the last but one release

how much space do databases take?

```
perl $ENSEMBL_CVS_ROOT_DIR/ensembl/misc-scripts/db/db-space.pl -host
compara4 -port 3306 -user ensadmin -password $ENSADMIN_PSW
```

Guidelines for the deletion of Databases

****Before dropping any of the databases, they should be backed up and zipped. At least the eHive tables should be backed on the production databases while research databases should be completely backed up.

- release database: keep only the last one (for HCs)
- production databases: keep the last production run of each pipeline on each species-set (e.g. keep the primate EPO ***and*** the mammal EPO ***and*** etc

Configuration file

- ☒ Update production_reg_ebi_conf.pl and check back into git:

✓ [Click for details](#)

Update the registry configuration file \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg_ebi_conf.pl that will be used throughout the release process.

Make sure to have edited the release numbers, added external core databases and fixed name prefixes.

The convention right now (rel.82) is to keep the merged release database on compara5.

DB connection details of production databases (families, nctrees, etc.) can be removed from the file until merge.

Schema preparation

All the schema changes must be ready by the handover of the core databases. See [SOP for API / schema changes](#) for the procedure about changing the schema

Update table.sql and create patch files

Here we need to prepare table.sql for the new schema and create the relevant patch files. The general procedure is defined in [SOP for API / schema changes](#).

The other Compara members may have already created patches, so check with them what's there. At the minimum, there should be the schema version increase.

✓ [Click here for details](#)

Update the \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/sql/table.sql file and create any patch files.

- ☒ Create a patch file for the schema_version
- ☒ Update the schema_version in table.sql and delete the other patch INSERT statements
- ☒ Check if any other patch files need creating by looking at the Declaration of Intentions and checking with other compara team members
- ☒ Add an INSERT statement for the new schema_version in table.sql and for any other new patches

Check the patch files

- ☒ The schema defined in the current table.sql must be obtainable by patching the previous database. There is a shell script to do the comparison:

✓ [Click here for details](#)

Run the script \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/production/schema_diff.sh (you may have to press "y" to validate the patches to apply to the old schema) and check the output. The script relies on several things:

- The API version declared by the Core API (in Bio::Ensembl::ApiVersion) has been updated
- The meta keys in the live database are correct (they should !)

The only allowed difference is that peptide_align_feature_XX tables are only found in the previous database, not the new one. The script writes 3 files to the current directory: old_schema.sql, patched_old_schema.sql, and new_schema.sql, and automatically runs sdiff. If you need to look at the diff later on, run this:

```
sdiff -w 200 -bs patched_old_schema.sql new_schema.sql
```

git commit table.sql and any patch files

How to patch a database to the latest schema

✓ [Click here to expand...](#)

- Use the following script to detect the what schema the database is on and to apply all the required patches to bring it to the latest schema.

(make sure to use the correct database name)

```
$ENSEMBL_CVS_ROOT_DIR/ensembl/misc-scripts/schema_patcher.pl
$(mysql-ens-compara-prod-1-ensadmin details script)
--database=ensembl_compara_${PREV_ENSEMBL_RELEASE}

$ENSEMBL_CVS_ROOT_DIR/ensembl/misc-scripts/schema_patcher.pl
$(mysql-ens-compara-prod-1-ensadmin details script)
--database=mateus_ensembl_ancestral_${PREV_ENSEMBL_RELEASE}
```

- ✓ Patch the previous production and ancestral databases using the above script
- ✓ Patch the master database (run the commands contained in the patch files individually, given that some patches may fail due to the master having an incomplete set of tables)

Master database

The Master database can be updated at any time but must be ready before the main pipelines start. For more information about its role, consult [Master database](#).

NCBI taxonomy data

The production team updates the ncbi_taxonomy database on livemirror just before the handover to us (please check that this has been done). We then need to update the tables on our master DB. The current (rel.90) master database is ensembl_compara_master on mysql-ens-compara-prod-1

- ✓ Update the ncbi_taxa_node and ncbi_taxa_name in the master database

✓ [Click here for details](#)

The ncbi_taxonomy database is located in `mysql://ensro@mysql-ensembl-mirror:4240/ncbi_taxonomy`

mysqldump

```
time db_cmd.pl $COMPARA_REG ncbi_taxonomy -reg_type taxonomy
--executable mysqldump --prepend --extended-insert --prepend
--compress ncbi_taxa_node ncbi_taxa_name | db_cmd.pl $COMPARA_REG
compara_master
```

It usually takes between 30 and 60 seconds.

- ✓ Load ensembl_aliases.sql onto the master database

✓ [Click here for details](#)

The script will report any discrepancies that need to be resolved ie any nodes which have been deleted from the ncbi_taxonomy

database but still have entries in the `ensembl_aliases.sql` file. Check if these have an entry in the `species_set_tag` table. If not, it is probably safe to delete them. Check with other compara team members.

load ensembl_aliases

```
db_cmd.pl $COMPARA_REG compara_master <
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/taxonomy/ensembl_aliases.sql
```

☒ Run the CheckTaxon healthcheck

▼ [Click here to expand...](#)

Run the CheckTaxon healthcheck early to find any discrepancies between the `ncbi_taxon_name` table and the core databases (information about how to set up the healthchecks can be found [here](#))

Run healthcheck

```
#cd to your local healthcheck git repo :
cd ensj-healthcheck/

# make sure you are using the right version of JAVA:
export JAVA_HOME=/software/jdk1.6.0_14

# if you need to recompile (submit to the farm, because you need
more memory than is available on the head) :
bsub -I ant clean jar

# run the healthchecks (submit to the farm, because you need more
memory than is available on the head) :
time bsub -I ./run-configurable-testrunner.sh
$(mysql-ens-compara-prod-1 details script) -d
ensembl_compara_master --release $CURR_ENSEMBL_RELEASE -t
org.ensembl.healthcheck.testcase.compara.CheckTaxon
```

Add new entries to compara master database

The current master database (e90) is called `ensembl_compara_master` on `mysql-ens-compara-prod-1`. You have to create new `genome_dbs` and `dnafrags` when there is a new assembly or a new species. Any new `genome_dbs`, `dnafrags` and `method_link_species_set_ids` need to be added before production starts.

☒ Add a new species / assembly (i.e. a `genome_db`)

▼ [Click here for details](#)

You will have to run this script once per new / updated species

Add genome_db

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/update_genom
e.pl --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl --compara compara_master --species "gadus_morhua"
--release --collection ensembl
```

If you know that the species won't be ready for this release but the next one, you need to remove "--release --collection ensembl". Next release, you'll have to rerun update_genome.pl with the --force option

Add the new genome_db_id to the confluence page [Release plans](#). This script may take a while if the species you are adding is new and has a lot of scaffolds. You can check the progress by counting *dnafrag* entries in the master database:

```
SELECT COUNT(*) FROM dnafrag;
```

Add the alias names and other tags to \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/taxonomy/ensembl_aliases.sql. Each extant species is anchored to the species tree at a certain taxon. This taxon must be described with two fields in the *ncbi_taxa_name* table. \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/taxonomy/place_species.pl helps in placing the new species in the current compara species tree and getting those tags.

check species/taxa from ensembl_production

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/taxonomy/place_specie
s.pl -master_url
mysql://ensro:mysql-ens-compara-prod-1:4485/ensembl_compara_master
-taxon_ids 9940,9361,7994,7918 -collection ensembl
```

For each taxon you need to add a '[ensembl timetree mya](#)' tag that defines the age of the taxon. If [ensembl timetree mya](#) is missing, go to the [TimeTree](#) URL given by the script and check whether the data really is missing, or if they changed the format of their URL / HTML pages.

For any new species, update the file \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/taxonomy/ensembl_aliases.sql to add the new tags and reload the file into the master database



Add in extra non-reference patches.

✓ [Click here for details](#)

This is currently done when a new patch for either human or mouse is released. This may have already been done, please ask.>

Details about the patches can be found here ftp://ftp.ncbi.nlm.nih.gov/genbank/genomes/Eukaryotes/vertebrates_mammals/Homo_sapiens/ e.g. for patch 11: ftp://ftp.ncbi.nlm.nih.gov/genbank/genomes/Eukaryotes/vertebrates_mammals/Homo_sapiens/GRCh37.p11/README

It is first necessary to find if any patches have been deleted or updated since alignments on these need to be deleted from the Compara database. This is done by running the find_assembly_patches.pl script on the new and previous release of the core database

Find assembly patches

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/find_assembly_patches.pl -compara
"mysql://ensro@mysql-ens-compara-prod-1:4485/ensembl_compara_master
" -new_core
"mysql://ensro@mysql-ens-sta-1:4519/homo_sapiens_core_88_38?group=core&species=homo_sapiens" -prev_core
"mysql://ensro@mysql-ensembl-mirror:4240/homo_sapiens_core_87_38?group=core&species=homo_sapiens"
```

Sample output

NEW patches

```
CHR_HG2088_PATCH 2006369791 2017-01-19 11:45:17
CHR_HSCHR19KIR_HG2394_CTG3_1 2006369821 2017-01-19 11:45:17
CHR_HSCHR19KIR_0019-4656-A_CTG3_1 2006369811 2017-01-19 11:45:17
CHR_HSCHR19KIR_7191059-1_CTG3_1 2006369829 2017-01-19 11:45:17
CHR_HSCHR19KIR_7191059-2_CTG3_1 2006369835 2017-01-19 11:45:17
CHR_HSCHR19KIR_0019-4656-B_CTG3_1 2006369831 2017-01-19 11:45:17
CHR_HSCHRX_3_CTG7 2006369841 2017-01-19 11:45:17
CHR_HG1708_PATCH 2006369799 2017-01-19 11:45:17
CHR_HG2236_PATCH 2006369783 2017-01-19 11:45:17
CHR_HSCHR19KIR_0010-5217-AB_CTG3_1 2006369827 2017-01-19 11:45:17
CHR_HSCHR17_11_CTG4 2006369809 2017-01-19 11:45:17
CHR_HSCHR19KIR_CA01-TB01_CTG3_1 2006369819 2017-01-19 11:45:17
CHR_HSCHR19KIR_HG2393_CTG3_1 2006369839 2017-01-19 11:45:17
CHR_HSCHR1_6_CTG3 2006369781 2017-01-19 11:45:17
CHR_HSCHR19KIR_CA04_CTG3_1 2006369833 2017-01-19 11:45:17
CHR_HG926_PATCH 2006369801 2017-01-19 11:45:17
CHR_HSCHR19KIR_CA01-TA01_1_CTG3_1 2006369813 2017-01-19 11:45:17
CHR_HSCHR19KIR_CA01-TB04_CTG3_1 2006369817 2017-01-19 11:45:17
CHR_HSCHR4_12_CTG12 2006369785 2017-01-19 11:45:17
CHR_HG2068_PATCH 2006369795 2017-01-19 11:45:17
CHR_HSCHR19KIR_502960008-1_CTG3_1 2006369825 2017-01-19 11:45:17
CHR_HG2046_PATCH 2006369805 2017-01-19 11:45:17
CHR_HG2285_HG106_HG2252_PATCH 2006369803 2017-01-19 11:45:17
CHR_HG2266_PATCH 2006369793 2017-01-19 11:45:17
CHR_HG2067_PATCH 2006369797 2017-01-19 11:45:17
CHR_HSCHR19KIR_CA01-TA01_2_CTG3_1 2006369815 2017-01-19 11:45:17
CHR_HSCHR19KIR_HG2396_CTG3_1 2006369837 2017-01-19 11:45:17
CHR_HG30_PATCH 2006369789 2017-01-19 11:45:17
CHR_HSCHR19KIR_502960008-2_CTG3_1 2006369823 2017-01-19 11:45:17
CHR_HSCHR5_8_CTG1 2006369787 2017-01-19 11:45:17
CHR_HSCHR17_3_CTG1 2006369807 2017-01-19 11:45:17
```

CHANGED patches

DELETED patches

DnaFragments to delete:

names:

dnafrag_ids:

Input for create_patch_pairaligner_conf.pl:

--patches

chromosome:CHR_HG1708_PATCH,chromosome:CHR_HSCHR19KIR_CA01-TA01_1_C
TG3_1,chromosome:CHR_HSCHR4_12_CTG12,chromosome:CHR_HSCHR19KIR_HG23
96_CTG3_1,chromosome:CHR_HSCHR17_3_CTG1,chromosome:CHR_HSCHR5_8_CTG
1,chromosome:CHR_HSCHR19KIR_7191059-1_CTG3_1,chromosome:CHR_HSCHR19
KIR_0019-4656-B_CTG3_1,chromosome:CHR_HSCHR19KIR_CA04_CTG3_1,chromo
some:CHR_HG926_PATCH,chromosome:CHR_HSCHR19KIR_CA01-TB04_CTG3_1,chr
omosome:CHR_HG2046_PATCH,chromosome:CHR_HG2285_HG106_HG2252_PATCH,c
hromosome:CHR_HG2067_PATCH,chromosome:CHR_HSCHR19KIR_HG2394_CTG3_1,
chromosome:CHR_HSCHR19KIR_0019-4656-A_CTG3_1,chromosome:CHR_HSCHR19
KIR_7191059-2_CTG3_1,chromosome:CHR_HSCHRX_3_CTG7,chromosome:CHR_HG
2236_PATCH,chromosome:CHR_HSCHR17_11_CTG4,chromosome:CHR_HSCHR1_6_C
TG3,chromosome:CHR_HSCHR19KIR_502960008-1_CTG3_1,chromosome:CHR_HG2
266_PATCH,chromosome:CHR_HSCHR19KIR_502960008-2_CTG3_1,chromosome:C
HR_HG2088_PATCH,chromosome:CHR_HSCHR19KIR_0010-5217-AB_CTG3_1,chrom

```
osome:CHR_HSCHR19KIR_HG2393_CTG3_1,chromosome:CHR_HSCHR19KIR_CA01-T  
B01_CTG3_1,chromosome:CHR_HG2068_PATCH,chromosome:CHR_HSCHR19KIR_CA  
01-TA01_2_CTG3_1,chromosome:CHR_HG30_PATCH
```

Copy the output to the Intentions for Release page as it will be needed to clean-up the alignments

In this case, there are 14 NEW patches, 1 CHANGED patch and 1 DELETED patch. They can be imported to / deleted from the master database by running `update_genome.pl` with the `--force` option

Add patches

```
perl  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/update_genom  
e.pl --reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --compara compara_master --species human --force
```

The steps for running the pairwise alignment pipeline for new patches can be found here:

`$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/pipelines/READMEs/pair_aligner_patches.txt`



Add in the new LRGs

▼ [Click here to expand...](#)

LRGs are needed by the Family pipeline, and have to be updated every release. This is done by running the `update_genome.pl` script on human with the `--force` option

```
perl  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/update_genom  
e.pl --reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --compara compara_master --species human --force
```

Note that the new LRGs may have already been loaded by the previous step (*add in the human patches*) as the same `update_genome.pl` command is run

To check if everything loaded OK, compare the output of the following queries:

```
db_cmd.pl --reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --reg_type core --reg_alias human -sql 'select  
count(*) from seq_region join coord_system cs  
using(coord_system_id) where cs.name="lrg" '  
  
db_cmd.pl --reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --reg_alias compara_master -sql 'select count(*)  
from dnafrag where coord_system_name="lrg" '
```

they should be the same.

Add method_link_species_set entries to compara master database

These are usually added by the people that need them, please check.

The release coordinator (or any team member) should create a new method_link_species_set in the master database before starting a new pipeline in order to get a unique method_link_species_set_id. Ideally they can be created before starting to build the new database although new method_link_species_sets can be added later on.

- ☒ Add dna method_link_species_set entries

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type LASTZ_NET --genome_db_id 90,142 --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl --compara compara_master
```

If you are sure that the mlss will be ready for the next release, you can add --release to the command-line. Otherwise you will have to mark the entry manually as OK before merging

- ☒ Add species tree method_link_species_set entry

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type SPECIES_TREE --force --collection "ensembl"
--source "ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release
```

- ☒ Add protein-trees method_link_species_set entry

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type PROTEIN_TREES --force --collection "default"
--source "ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type PROTEIN_TREES --force --collection "murinae"
--source "ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release
```

- ☒ Add ncRNA-trees method_link_species_set entry

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type NC_TREES --force --collection "ensembl" --source
"ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release
```

- ☒ Add family method_link_species_set entry

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type FAMILY --force --collection "ensembl" --source
"ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release
```

- ☒ Add pairwise ortholog method_link_species_set entry

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type ENSEMBL_ORTHOLOGUES --force --pw --collection
"ensembl" --source "ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release 2>
create_mlss.${CURR_ENSEMBL_RELEASE}.ENSEMBL_ORTHOLOGUES.err >
create_mlss.${CURR_ENSEMBL_RELEASE}.ENSEMBL_ORTHOLOGUES.out
```

Inspect the out and err files for errors

- ☒ Add singleton paralog method_link_species_set entry

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/create_mlss.pl
--method_link_type ENSEMBL_PARALOGUES --force --sg --collection
"ensembl" --source "ensembl" --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg
_ebi_conf.pl -compara compara_master --release 2>
create_mlss.${CURR_ENSEMBL_RELEASE}.ENSEMBL_PARALOGUES.err >
create_mlss.${CURR_ENSEMBL_RELEASE}.ENSEMBL_PARALOGUES.out
```

Inspect the out and err files for errors

- ☒ Reset the URL of reused mlss_ids
In case the same mlss_id can be reused, the pipeline will probably complain that there is already a URL attached to it. You need to reset these URLs

```
UPDATE method_link_species_set SET url = "" WHERE
method_link_species_set_id IN (<LIST_OF_mlss_ids>);
```

Usually, this happens when there are no new assemblies, in which case you need to give the mlss_id of the Family, ncRNA-tree and protein-tree pipelines

Add new species to phylogenetic tree

✓ Add new species to phylogenetic tree

✓ [Click here for details](#)

The best GUI for editing trees in this way is [TreeGraph2](#)

1. Open the tree you want to edit (found at `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/species_tree.*`)
2. Click on a node/branch to highlight it, right click and select 'Add node'
3. Repeat for all new species
4. Export the tree to newick format
5. Check files and `git commit`

Final edits to compara master database

This runs once all the species have been added / updated.

✓ Compare the staging servers to the master database

✓ [Click here to expand...](#)

This script will list the genomes of the staging servers, and compare them to the master database.

```
# Run once in dry-run mode to see the changes
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/update_master_db.pl --reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg_ebi_conf.pl --compara compara_master --dry_run
```

**** note:** The above might throw a warning about not finding any species name in the ancestral db, this is expected and should not break the test!!

Some things (like the genebuild date is different) can be directly changed by the script (use the `--nodry-run` option instead). If there is a new assembly / species on the staging servers that is not yet in the master database, run `update_genome.pl` (see above) to add it, and re-run `update_master_db.pl` to make sure things are solved.

✓ Healthcheck the master database

✓ [Click here to expand...](#)

See [JAVA Healthchecks](#) and use the `ComparaMaster` group

```
#cd to your local repo of healthcheck
cd ensj-healthcheck/
$ENSEMBL_CVS_ROOT_DIR/ensj-healthcheck/run-configurable-testrunner.
sh $(mysql-ens-compara-prod-1 details script) -d
ensembl_compara_master --release $CURR_ENSEMBL_RELEASE -g
ComparaMaster
```

Production

Members must be loaded

As of release 90, members are preloaded before any production pipelines are run. Use the following pipeline to load the members:

```
init_pipeline.pl Bio::Ensembl::Compara::PipeConfig::LoadMembers_conf
--collection ensembl
```

Remember to pass the `member_db` parameter to the production pipelines at initialisation, or update the param in the config files!!

Main site

List the pipelines that have to be run on the [Intentions for Release](#) page and schedule the work with the rest of the team.

GRCh37 site [\[edit\]](#)

For the next 10 years or so, we'll maintain a special archive site for the GRCh37 assembly: <http://grch37.ensembl.org>

This page explains the procedure to update the Compara databases on it. This has to be done every release and is coordinated with the other teams.

Make sure both the `ensembl_compara` and the `ensembl_ancestral` databases have been patched to the latest schemas before handing them over !

Connection details

The web-team has a Confluence page with all the gory details. To summarize:

- Public MySQL server: `ensemldb.ensembl.org:3337`
- Internal MySQL server (where we need to update the databases): `ens-staging-grch37` in `e!84`, but it may be different in future releases

Case 1: no data updates

- Once we have committed our patches for the release, the production team does a nightly pull and patch all databases automatically. Including so we do not have to do anything on our end anymore as off Rel89

Case 2: data updates

Need to identify which data can / have to be recomputed, and coordinated with the other teams. Changes **must** be declared as intentions (select "GRCh37" in "Site type") and the databases patched to the latest schema

Past updates:

▼ [Click here to expand...](#)

- **NOT SCHEDULED** Upgrade of the protein trees / homologies, to merge with TreeFam. This will impact BioMart, etc.
- **DONE** Fix the CAFE data. Actually done at a data update, but I can't remember which one
- **DONE** e87 - Family pipeline (HMMs instead of `blastp+mcl`)
- **DONE** e87 - Added GOC and WGA scores
- **DONE** e87 - Cleaned-up human self-alignments (should only be between primary regions)
- **DONE** e87 - Added JSON blobs for CAFE and `exon_boundaries`

End of production window

Create Release Database

Create the new database for the new release and add it to your registry configuration file. Use the `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/sql/table.sql` file to create the tables and populate the database with the relevant primary data and genomic alignments that can be reused from the previous release. This can be done with the `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/populate_new_database.pl` script. It requires the master database, the previous released database and the fresh new database with the tables already created. The script will copy relevant data from the master and the old database into the new one.

- ☒ >Create new database
 - [Click here for details](#)

Create database

```
db_cmd.pl $COMPARA_REG compara_curr -sql "CREATE DATABASE "  
db_cmd.pl $COMPARA_REG compara_curr <  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/sql/table.sql
```

- ☒ Populate the new database
 - [Click here for details](#)

Before you start copying, make a dry run of the `populate_new_database.pl` with `-intentions` flag to review the list of `mlss_ids` to be copied:

populate_new_database intentions

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/populat  
e_new_database.pl --reg-conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/product  
ion_reg_ebi_conf.pl \  
--master compara_master --old compara_prev --new compara_curr  
--intentions > populate_new_database.intentions
```

This normally takes less than a minute and produces a long list.

If you believe some of the MLSS or SS entries should NOT be copied, connect to the master database and change the `last_release` of the unwanted entries to the previous release number. Conversely, if you want to prepare some new MLSS or SS entries to be copied (e.g. the newly run pipelines), change the `first_release` of the wanted entries to the current release number.

NB: OLD INSTRUCTIONS FOR PRE-first_release/last_release API: *There are cases where the mlss does not change but the underlying data does, e.g. the "patch-to-ref" alignment (H.sap-H.sap lastz-patch and M.mus-M.mus lastz-patch). These have a mlss_id of 556 (H.sap) and 624 (M.mus) and are currently set in the skip_mlss. If there are no new patches, this needs to be removed to allow the existing data to be copied. If there are new patches, please ensure the 'skip_mlss' is set in the meta table. However, the entry in the method_link_species_set table will not be copied and will need to be added manually.*

Start the copying:

populate_new_database

```
bsub -q production-rh7 -R "select[mem>20000]
rusage[mem=20000]" -M20000 -Is bash
time
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/populat
e_new_database.pl \
--reg-conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/product
ion_reg_ebi_conf.pl --master compara_master --old compara_prev
--new compara_curr > populate_new_database.out
```

▼ [Click here for run times](#)

rel. 88: 640m18.007s

took 351m23.029s (~5.85 hours) for rel.87

took 3 hours for rel.pre57 (copied from rel.56)

took 3 hours for rel.57 (copied from rel.pre57)

took 2:15 hours for rel.58 (copied from rel.57)

took 2:09 hours for rel.59 (copied from rel.58)

took 3 hours for rel. 60 (copied from rel.59)

rel.64: 2.6h

rel.65: 2.5h

rel.66: 4.8h

rel.67: 2.1h (launched from compara3)

rel.68: 1h40m (run on compara3)

rel.69: 2.5h

rel.70: ~3.5h (compara1 was slow)

rel.71: 4.1h (compara3)

rel.72: 5.1h (compara3)

rel.73: 5.5h (compara2)

rel.74: 2h:3' (compara3)

rel.75: 5.5h (compara5)

rel.77: 9.7h (compara5)

rel.78: 6.0h (compara4)

rel.79:

rel.80:

rel.81: 6h (compara5)

rel.82: 5.5h (compara5)

rel.83: 4.8h (compara5)

rel.85: 7.5h (compara5)

If new method_link_species_sets are added in the master after this, you use this script again to copy the new relevant data. In such case, you will have to:

- skip the old_database in order to avoid trying to copy the dna-dna alignments and syntenies again
- empty ncbi_taxa_name before running

populate_new_database from master only

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/populat
e_new_database.pl \
--reg-conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/product
ion_reg_conf.pl --master compara_master --new compara_curr
```


- ☒ Delete any pairwise alignments on non-reference patches that have been DELETED or UPDATED.

▼ [Click here for details](#)

Find the output of `find_assembly_patches.pl` script that you ran previously (usually for Human and Mouse) and combine their "Dnafrags to delete" into one common list:

delete patches

```
DNAFRAGS_2_DELETE="(14025314,14025313)"
```

```
db_cmd.pl $COMPARA_REG compara_curr -sql "SELECT count(*) FROM
genomic_align ga1, genomic_align ga2, genomic_align_block gab WHERE
ga1.genomic_align_block_id = ga2.genomic_align_block_id AND
ga1.genomic_align_id != ga2.genomic_align_id AND
ga1.genomic_align_block_id = gab.genomic_align_block_id AND
ga1.dnafrag_id in $DNAFRAGS_2_DELETE"
```

```
db_cmd.pl $COMPARA_REG compara_curr -sql "DELETE ga1, ga2, gab FROM
genomic_align ga1, genomic_align ga2, genomic_align_block gab WHERE
ga1.genomic_align_block_id = ga2.genomic_align_block_id AND
ga1.genomic_align_id != ga2.genomic_align_id AND
ga1.genomic_align_block_id = gab.genomic_align_block_id AND
ga1.dnafrag_id in $DNAFRAGS_2_DELETE"
```

- ☒ Run healthchecks on the release database

▼ [Click here for details](#)

Run the healthchecks to make sure the the release database is consistent after the initial population of data.

Click [here](#) for how to setup and run the healthchecks

Run the `compara_external_foreign_keys` healthcheck

healthcheck

```
cd $ENSEMBL_CVS_ROOT_DIR/ensj-healthcheck
```

```
# make sure you are using the right version of JAVA:
export JAVA_HOME=/software/jdk1.6.0_14
```

```
# if you need to recompile (submit to the farm, because you need
more memory than is available on the head) :
bsub -I ant clean jar
```

```
# some tests need more memory than the farm3's default:
time bsub -q production-rh7 -M8000 -R"select[mem>8000]
rusage[mem=8000]" -I ./run-configurable-testrunner.sh
$(mysql-ens-compara-prod-1 details script) -d ensembl_compara_88
--release $CURR_ENSEMBL_RELEASE -g ComparaShared
```

At this point its OK to have some unused method_links since they will be removed later. But should NOT be removed now since they may still be used by the merging.

▼ [Click here for run times](#)

rel.88 1:15m

rel.83: 13 minutes, 2 expected complaints (CheckSpeciesSetSizeByMethod may complain about Human-on-Human lastz-new and ForeignKeyMasterTables will complain about empty MethodLink entries (this will be deleted later in the merging process))

rel.85: 20mins

and correct any newly detected problems

☒ Update the declared intention with removed / deprecated methods

▼ [Click for details](#)

We need to generate the list of methods exported by the objects / adaptors on the master and release/{n-1} branches, and compare (diff) them.

Check deprecated / removed methods

```
# on master/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " *pm | sort > ~/MASTER_objects
# on the previous release/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " *pm | sort > ~/RELEASE87_objects
sdiff -w 200 -bs ~/RELEASE87_objects ~/MASTER_objects | less

# Let's do the same for the adaptors
# on master/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " DBSQL/*pm | sort > ~/MASTER_adaptors
# on the previous release/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " DBSQL/*pm | sort > ~/RELEASE87_adaptors
sdiff -w 200 -bs ~/RELEASE86_adaptors ~/MASTER_adaptors | less
```

In both cases, make sure the methods are really removed, and not moved to a base / sub class, etc

Merge DNA data

NOTE: All the runs of copy_data.pl (except the last one) should have the flag "-re_enable 0" to avoid recomputing the indices in the end of each run.

Running "-re_enable 1" will add *at least 2 hours* (rel.82) to the merging time (but it is necessary in the final product) so make sure you only do it once.

☒ Pairwise alignments: LASTZ_NET (and, formerly, BLASTZ_NET or TRANSLATED_BLAT_NET)

NOTE : For merging pw alignments involving haplotypes, go to the next point

▼ [Click here for details](#)

These data are usually in separate production databases. You can copy them using the \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl script. This script requires write access to the production database if the dnafrag_ids need fixing. Use the flag -re_enable 0 on all calls apart from the last one to avoid recomputing the indices.

Also, check first_release of these databases. In case it hasn't been set, you need to do it **now** on **both** the production database and the master database,

Example:

copy_data

```
# for each source URL: first plug in the --from_url and add
--dry_run to check that the script has found the right MLSS:
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl
--reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_conf.pl --to_reg_name compara_curr --method_link_type LASTZ_NET
--re_enable 0 --from_url
mysql://ensadmin:${ENSADMIN_PSW}@compara4/sf5_ggal_falb_lastz_73
--dry_run

# if happy, remove the --dry_run flag and run it again,
preferably on the farm:
bsub -q yesterday -R "select[mem>5000] rusage[mem=5000]" -M5000 -I
time
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl
--reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_conf.pl --to_reg_name compara_curr --method_link_type LASTZ_NET
--re_enable 0 --from_url
mysql://ensadmin:${ENSADMIN_PSW}@compara4/sf5_ggal_falb_lastz_73
```

✓ The curious case of LASTZ_PATCH alignments. There is **always** something to copy, even if there are no new patches

✓ [Click here for details](#)

You will also have to copy Human_ref_vs_Human_patches and Mouse_ref_vs_Mouse_patches LASTZ_PATCH alignments, but mind the source:

If there were new patches, you'll import them in a way similar to other LASTZ:

copy_data

```
#First run the following pipeline to import the alignments between
patches / haplotypes and primary regions.

init_pipeline.pl
Bio::Ensembl::Compara::PipeConfig::ImportPatchAlignmentsToRef_conf
-host comparaX

#then
# note the method_link_type is LASTZ_PATCH !
bsub -q yesterday -R "select[mem>5000] rusage[mem=5000]" -M5000 -I
time
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl
--reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl --to_reg_name compara_curr --method_link_type
LASTZ_PATCH --re_enable 0 --from_url
<the_url_of_the_pipeline_db_from_above>
```

If there were no new patches, you will still have to copy them from `compara_prev`, since `LASTZ_PATCH` alignments are automatically skipped by `populate_new_database.pl` script. You simply have to refer to the previous database as the source:

copy_data

```
# note the method_link_type is LASTZ_PATCH !
bsub -q yesterday -R "select[mem>5000] rusage[mem=5000]" -M5000 -I
time
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl
--reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl --to_reg_name compara_curr --method_link_type
LASTZ_PATCH --re_enable 1 --from_reg_name compara_prev
```

- ☒ Pairwise alignments: non-reference patches for the high coverage `LASTZ_NET` alignments. This is to be used when merging pairwise alignments involving haplotypes.

✓ [Click here for details](#)

This step is now very similar to the previous.

Do not forget the `--merge` option.

Also, if it's the last one you might want to switch keys back on

copy_data --merge --patch_merge

```
# first plug in the --from_url and add --dry_run to check that the
script has found the right MLSS:
bsub -R "select[mem>5000] rusage[mem=5000]" -M5000 -I time
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl
--reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl --to_reg_name compara_curr --method_link_type
LASTZ_NET --method_link_type BLASTZ_NET --method_link_type
TRANSLATED_BLAT_NET --re_enable 0 --merge --from_url
mysql://ensro@mysql-ens-compara-prod-1.ebi.ac.uk:4485/carlac_lastz_
human_patches_88 --dry_run
```

```
# if happy, remove the --dry_run flag and run it again,
preferably on the farm:
bsub -R "select[mem>5000] rusage[mem=5000]" -M5000 -I time
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl
--reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl --to_reg_name compara_curr --method_link_type
LASTZ_NET --method_link_type BLASTZ_NET --method_link_type
TRANSLATED_BLAT_NET --re_enable 1 --merge --from_url
mysql://ensro@mysql-ens-compara-prod-1.ebi.ac.uk:4485/carlac_lastz_
human_patches_88
```

- ☒ Multiple alignments: `PECAN`, `EPO`, `EPO_LOW_COVERAGE`, `GERP_CONSTRAINED_ELEMENT`, `GERP_CONSERVATION_SCORE`

✓ [Click here for details](#)

These data are usually in separate production databases. You can copy them using the `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl` script. This script requires write access to the production database if the `dnafrag_ids` need fixing or the data must be copied in binary mode (this is required for conservation

scores).

Some alignments produce conservation scores and constrained elements (check the [Release plans](#)) and these need to be copied separately.

eg

copy_data multiple alignment

```
bsub -q yesterday -R "select[mem>5000] rusage[mem=5000]" -M5000 \  
-I time  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl \  
\   
--reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --to_reg_name compara_curr \  
--method_link_type EPO --method_link_type EPO_LOW_COVERAGE  
--method_link_type PECAN \  
--method_link_type GERP_CONSTRAINED_ELEMENT --method_link_type  
GERP_CONSERVATION_SCORE \  
--from_url  
mysql://ensadmin:${ENSADMIN_PSW}@compara2/sf5_epo_low_8way_fish_71  
-re_enable 0
```

EPO alignments produce ancestral sequences and a separate core database which must also be copied. See below.

✓ [Click here for run times](#)

```
rel 71.  
2m: kb3_hsap_ggal_lastz_71 mlss_id=632  
1m kb3_mmus_ggal_lastz_71 mlss_id=633  
1m kb3_ggal_mgap_lastz_71 mlss_id=634  
1m kb3_ggal_xtro_tblast_71 mlss_id=638  
1m kb3_hsap_ggal_tblast_71 mlss_id=637  
1m sf5_olat_gmor_lastz_71 mlss_id=625  
3m kb3_pecan_20way_71 mlss_id=630  
4m kb3_pecan_20way_71 mlss_id=631  
35m kb3_pecan_20way_71 mlss_id=50045  
3m sf5_compara_epo_6way_71 mlss_id=548  
1m sf5_olat_onil_lastz_71 mlss_id=626  
1m sf5_olat_xmac_lastz_71 mlss_id=627  
1m sf5_epo_low_8way_fish_71 mlss_id=628  
2m sf5_epo_low_8way_fish_71 mlss_id=629  
9m sf5_epo_low_8way_fish_71 mlss_id=50044  
1m kb3_ggal_drer_tblast_71 mlss_id=639  
1m kb3_ggal_csav_tblast_71 mlss_id=640  
1m sf5_ggal_acar_lastz_71 mlss_id=636  
91m sf5_ggal_tgut_lastz_7 mlss_id=635 (re-enable 1)  
93m sf5_compara_epo_3way_birds_71 mlss_id=641 (re-enable 1)  
14m sf5_compara_epo_3way_birds_71 mlss_id=642 (re-enable 1)  
16m sf5_compara_epo_3way_birds_71 mlss_id=50046 (re-enable 1)
```

✓ Check the keys have been re-enabled

✓ [Click here for details](#)

Use `mysqlshow` to highlight if the table still has disabled keys. The text "disabled" will be shown in the Comment column if the key is disabled. An empty Comment column indicates the keys are enabled.

`mysqlshow` interprets any underscores in the last argument as a wildcard so to get round this, we need to use % as the last argument.

mysqlshow

```
db_cmd.pl $COMPARA_REG compara_curr --executable mysqlshow --  
--keys genomic_align_block %  
db_cmd.pl $COMPARA_REG compara_curr --executable mysqlshow --  
--keys genomic_align %  
db_cmd.pl $COMPARA_REG compara_curr --executable mysqlshow --  
--keys genomic_align_tree %  
db_cmd.pl $COMPARA_REG compara_curr --executable mysqlshow --  
--keys conservation_score %  
db_cmd.pl $COMPARA_REG compara_curr --executable mysqlshow --  
--keys constrained_element %
```

If there are still tables with keys disabled run the following on them:

```
db_cmd.pl $COMPARA_REG compara_curr -sql "ALTER TABLE <table_name>  
ENABLE KEYS";
```



Synteny

[Click here for details](#)

First make sure the entries in \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_reg_ebi_conf.pl file point at the latest (staging) versions of the core databases.

Before running the code... Ensure that you check the syntenic coverage and if it is less than 1%, it must be deleted from mlss, mlss_tag, dnafrag_region and syntenic_region tables. **This should be automated in the syntenic pipeline by release 84.

Example

load syntenic data

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl  
--reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --to_reg_name compara_curr --method_link_type  
SYNTENY --from_url mysql://ensro@compara1/cc21_syntenic_83 --dry_run  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/copy_data.pl  
--reg_conf  
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r  
eg_ebi_conf.pl --to_reg_name compara_curr --method_link_type  
SYNTENY --from_url mysql://ensro@compara1/cc21_syntenic_83
```



Build a new ancestral sequence core database

[Click here for details](#)

Putting together the database of ancestral sequence is now done using a dedicated Hive-Core mini-pipeline.

Check you have the most recent core checkout ie the correct schema and patch files are added to the meta table.

Go to \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/modules/Bio/Ensembl/Compara/PipeConfig and open the PipeConfig file AncestralMerge_conf.pm .

Make sure you have edited/checked the following:

- 1) current release number
- 2) names and locations of current and previous ancestral core databases

3) the table of ancestral sequence sources in the second analysis (some entries might point to the previous release ancestral database, some will be new)

For (3), you can run the following query on your release database and on the previous database: (NB: method_link_id=13 is equivalent to method_link_type = "EPO")

EPO query

```
SELECT * FROM method_link_species_set WHERE method_link_id = 13;
```

The new mlss_id should be attached to their production database:

'641' => 'mysql://ensadmin:\$ENSADMIN_PSW@compara3/sf5_3birds_ancestral_sequences_core_71'

The mlss_id that are reused should be linked to the previous database

'505' => \$self->o('prev_ancestral_db'),

The current (as of rel.75) list of ancestral alignments are:

5 teleost fish
6 primates
4 sauropsids ("birds")
15 eutherian mammals

Save the changes, exit the editor and run init_pipeline.pl with this file:

init_pipeline

```
init_pipeline.pl AncestralMerge_conf.pm -host compara5
```

Then run both -sync and -loop variations of the beekeeper.pl command suggested by init_pipeline.pl . This pipeline will merge the separate ancestral core sources into ensembl_ancestral_{rel_number}.

You may want to check the msg table for errors and have a look at the result of the merger:

Which Ancestral sequences do we have?

```
SELECT left(name,12) na, count(*), min(seq_region_id),  
max(seq_region_id), max(seq_region_id)-min(seq_region_id)+1 FROM  
seq_region GROUP BY na;
```

If everything is ok, measure the time:

how much time did running of the pipeline take?

```
call time_analysis('%')
```

✓ [Click here for run times](#)

rel.87: 29.5min

rel.86: 58min

rel.67: 20min

rel.71: 20min

rel.75: 21min

Then drop hive-specific tables:

drop hive tables

```
CALL drop_hive_tables;
```

Make sure all tables are myISAM.

```
SHOW TABLE STATUS where engine != 'MyISAM';
```

or, if no new multiple alignments were run, copy it over from the previous release

✓ [Click here for details](#)

Create a new database for ancestral sequences:

```
db_cmd.pl $COMPARA_REG ancestral_curr -reg_type core -sql 'CREATE
DATABASE '
```

Copy over the data from the previous release:

```
time db_cmd.pl $COMPARA_REG ancestral_prev -reg_type core
-executable mysqldump | db_cmd.pl $COMPARA_REG ancestral_curr
-reg_type core
# rel.88: 54m44.503s
# rel.85: 40mins
# took 45 minutes in rel.81
# took 42 minutes in rel.82
# took 38 minutes in rel.83
# took 38 minutes in rel.84
# took 54 minutes in rel.89
```

Patch the database to the current release by applying the relevant patches from \$ENSEMBL_CVS_ROOT_DIR/ensembl/sql or use a schema patcher script.

```
$ENSEMBL_CVS_ROOT_DIR/ensembl/misc-scripts/schema_patcher.pl
--host=compara5 --user=ensadmin --pass=${ENSADMIN_PSW}
--database=lg4_ensembl_ancestral_${CURR_ENSEMBL_RELEASE}
```

If patches were applied, make sure you have both analyzed and optimized the tables:

```
db_cmd.pl $COMPARA_REG ancestral_curr -reg_type core -executable
mysqlcheck -- --analyze --verbose
db_cmd.pl $COMPARA_REG ancestral_curr -reg_type core -executable
mysqlcheck -- --optimize --verbose
```

Merge GeneTrees+Families+NCTrees+PatchProjectionsAsHomologies

- ☒ Check that Protein-trees have been run and handed-over ("*Compara hands over Homologies*" date)
- ☒ Check that ncRNA-trees have been run
- ☒ Check that Families have been run
- ☒ Check that LRGs were included in the Families
- ☒ Check that the Orthology QC WGA-score pipeline have been run
- ☒ Run the Hive pipeline (*EnsemblMergeDBsIntoRelease_conf*) to merge tables from all the four products into the release database
 - ▼ [Click here for details](#)
 - Go to \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/modules/Bio/EnsEMBL/Compara/PipeConfig/Example and open the PipeConfig file EnsemblMergeDBsIntoRelease_conf.pm

It has a 'urls' hash where you will have to change the names of the databases and possibly their locations:

master_db - is the main compara master
 prev_rel_db - should point to the previous release database
 curr_rel_db - should point to the current release database being merged into (not the Hive pipeline database, but purely Compara schema product)

protein_db - should point to the current GeneTrees pipeline database
 family_db - should point to the current Families pipeline database
 ncna_db - should point to the current ncRNAtrees pipeline database
 mouse_db - should point to the current Mouse-strasins pipeline database.

Also choose the server to run the merging pipeline on (you don't need a lot of resources or memory, as it is purely Hive book-keeping) and set the 'host' default_option.

Save the changes, exit the editor, init and run the merging pipeline with this file:

running the merging pipeline

```
init_pipeline.pl EnsemblMergeDBsIntoRelease_conf.pm
beekeeper.pl ... -sync
beekeeper.pl ... -loop
```

This pipeline will merge all the protein-side products into the release database.

**** NOTE: you may have to do some cleaning of the mouse strain db e.g. deleting some duplicated mlss ids.

▼ [Click here for times](#)

rel.73 was the first experimental run, code had to be fixed, servers had to be reconfigured, so merging took one whole working day.

In the merging database run: call time_analysis('%');

rel.75 : 5 hours
 rel.76 : 5.6 hours

...

rel.82 : 6.1 hours

- ☒ Load the species-trees (needed for the Species-tree view)

```
$ init_pipeline.pl
Bio::EnsEMBL::Compara::PipeConfig::LoadSpeciesTrees_conf
-compara_alias_name compara_curr -host mysql-ens-compara-prod-1 -port
4485
# Then run beekeeper as suggested by init_pipeline.pl
```

Note: the last analysis of this pipeline failed in rel.82 (all 4 jobs of this analysis) trying to insert duplicated entries into MLSS_tag table, but all the data was there, so I just carried on. same in rel.85

☒ Run the Bio::EnsEMBL::Compara::PipeConfig::Example::EnsemblPostHomologyMerge_conf pipeline:

✓ [Click here to expand...](#)

This pipeline combines all the pipelines we have to run at this stage: ImportAltAlleGroupsAsHomologies_conf, GeneMemberHomologyStats_conf, UpdateMemberNamesDescriptions_conf and HighConfidenceOrthologs_conf. Each of these can be disabled with a flag, and they're also all available as a standalone pipelines.

- [ImportAltAlleGroupsAsHomologies_conf](#). The core databases contain "alt-allele groups" which groups equivalent genes between the chromosomes and their patches, For Web to be able to display these data in RegionComparison, the groups (links) must be loaded in the Compara database as homologies under the description "alt-allele".
- [GeneMemberHomologyStats_conf](#). It populates the gene_member_hom_stats table (absence/presence of gene-tree, number of orthologues, etc) which is used by Web to grey out the menu items. This part comes pre-seeded with two jobs (one for the *default* collection and one for the *murinae* collection).
- [UpdateMemberNamesDescriptions_conf](#). The production team projects gene names and descriptions (and other xrefs) following our orthology calls. This happens *after* we handover the homologies, so the members that we have at this stage in the database have *old* names. This pipeline pulls the new names and descriptions.
- [HighConfidenceOrthologs_conf](#). This pipeline marks some orthologues as high-confidence based on their % identity, GOC and WGA scores. Because GOC is not computed on ncRNAs, the pipeline has two stream of jobs: one for protein-coding genes and one for ncRNAs. ⚠ This needs the WGA scores to be in !

init_pipeline.pl Bio::EnsEMBL::Compara::PipeConfig::Example::EnsemblPostHomologyMerge_conf -compara_db mysql://ensadmin:xxxx@mysql-ens-compara-prod-1:4485/ensembl_compara_88

☐ Drop the three databases used for merging.

✓ [Click here for details](#)

```
# After you are happy about the result of protein side merging you
can drop the "YourName_homology_projections_ThisRelease" database.
$ db_cmd.pl -url
mysql://ensadmin:${ENSADMIN_PSW}@compara5/lg4_homology_projections_
${CURR_ENSEMBL_RELEASE} -sql 'drop database'

# same for the LoadSpeciesTrees database:
$ db_cmd.pl -url
mysql://ensadmin:${ENSADMIN_PSW}@compara5/lg4_load_species_trees_${
CURR_ENSEMBL_RELEASE} -sql 'drop database'

# same for the MergeDBsIntoRelease database:
$ db_cmd.pl -url
mysql://ensadmin:${ENSADMIN_PSW}@compara5/lg4_pipeline_dbmerge_${CU
RR_ENSEMBL_RELEASE} -sql 'drop database'
```

☒ git commit the changes to the PipeConfig files that you have made.

Final database checks

- ✓ Remove redundant method_link entries
 - ✓ [Click here for details](#)

In most cases they can be removed, but check with other members of Compara. Remove redundant method_link entries

method_link entries

```
-- prepend with:      db_cmd.pl -reg_conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl -reg_alias compara_curr -sql
```

```
SELECT ml.* FROM method_link ml LEFT JOIN method_link_species_set
mlss USING(method_link_id) WHERE mlss.method_link_id IS NULL;
DELETE ml FROM method_link ml LEFT JOIN method_link_species_set
mlss USING(method_link_id) WHERE mlss.method_link_id IS NULL;
```

```
note**** we deleted 18 mlss_ids in rel.83, rel.86 & rel.88, rel.89;
19 in rel. 90
```

- ✓ Check that all the schema patches have been declared and applied.
 - ✓ [Click here for details](#)

If unsure, recheck the current schema against the previous schema. See Check the patch files for details

Run the healthchecks

- ✓ Update the code
 - ✓ [Click here for details](#)

The healthchecks are written in java and need to be recompiled after a git pull.

compile healthchecks

```
cd $ENSEMBL_CVS_ROOT_DIR/ensj-healthcheck
export JAVA_HOME=/software/jdk1.6.0_14
git pull
bsub -I ant clean jar
```

We don't need to configure a database.properties any more. Everything is done from the command line

- ✓ Run the healthchecks for ancestral database
 - ✓ [Click here for details](#)

```
time bsub -M8000 -R"select[mem>8000] rusage[mem=8000]" -I
./run-configurable-testrunner.sh $(mysql-ens-compara-prod-1 details
script) -d mateus_ensembl_ancestral_88 --release
$CURR_ENSEMBL_RELEASE -g ComparaAncestral
```

It should take less than a minute (if the tables are analyzed / optimized) and usually complains about 1 thing that you can ignore:

```
org.ensembl.healthcheck.testcase.generic.AssemblySegregion [Team
responsible: GENEBUILD]
mm14_ensembl_ancestral_80: 0 rows found in assembly table
```

If healthcheck indicates that tables need to be analysed, follow instructions here: [Analyze / Optimize the databases](#)

- ✓ Update the max_alignment_length IF NECESSARY.

✓ [Click here for details](#)

Check that the max_alignment_lengths have been computed.

update max_alignment_length

```
time bsub -I ./run-configurable-testrunner.sh
$(mysql-ens-compara-prod-1 details script) -d ensembl_compara_88
--release $CURR_ENSEMBL_RELEASE -t
org.ensembl.healthcheck.testcase.compara.MLSSTagMaxAlign
```

If not (the healthcheck is failing), you can repair it by adding the --repair flag:

update max_alignment_length

```
time bsub -I ./run-configurable-testrunner.sh -h compara5 -d
sf5_ensembl_compara_77 --release $CURR_ENSEMBL_RELEASE -t
org.ensembl.healthcheck.testcase.compara.MLSSTagMaxAlign --repair 1
--user ensadmin --password $ENSADMIN_PSW
```

- ✓ Update the alignment mlss_id of the conservation score IF NECESSARY

✓ [Click here for details](#)

update conservation score mlss_id

```
time bsub -I ./run-configurable-testrunner.sh
$(mysql-ens-compara-prod-1 details script) -d ensembl_compara_88
--release $CURR_ENSEMBL_RELEASE -t
org.ensembl.healthcheck.testcase.compara.MLSSTagGERPMSA
```

If the healthcheck is failing, you can repair it by adding the --repair flag:

update conservation score mlss_id

```
time bsub -I ./run-configurable-testrunner.sh -h compara5 -d
sf5_ensembl_compara_77 --release $CURR_ENSEMBL_RELEASE -t
org.ensembl.healthcheck.testcase.compara.MLSSTagGERPMSA --repair 1
--user ensadmin --password $ENSADMIN_PSW
```

- ✓ Run the ComparaAll group of healthchecks on the release database. NOTE: If pressed for time, this test can be divided into 2 separate tests that can be run simultaneously to save some time. You just have to substitute "ComparaAll" in the code block below to "ComparaHomology" in one run and "ComparaGenomic" in the other run

✓ [Click here for details](#)

The 'stdbuf -o0' is a trick to prevent the pipe from buffering the output, since in addition to storing it we also want to examine the output visually.

compara_external_foreign_keys

```
time bsub -M8000 -R"select[mem>8000] rusage[mem=8000]" -I stdbuf
-o0 ./run-configurable-testrunner.sh $(mysql-ens-compara-prod-1
details script) -d ensembl_compara_88 --release
$CURR_ENSEMBL_RELEASE -g ComparaAll | tee
healthchecks_after_merge.txt
```

✓ Run the ControlledComparaTables group of healthchecks on the release database

✓ [Click here for details](#)

The 'stdbuf -o0' is a trick to prevent the pipe from buffering the output, since in addition to storing it we also want to examine the output visually.

compara_external_foreign_keys

```
time bsub -M8000 -R"select[mem>8000] rusage[mem=8000]" -I stdbuf
-o0 ./run-configurable-testrunner.sh $(mysql-ens-compara-prod-1
details script) -d ensembl_compara_89 --release 89
--compara_master.database ensembl_compara_master -g
ControlledComparaTables | tee
healthchecks_controlled_tables_after_merge.txt
```

✓ Run the ComparaSanity group of healthchecks on the release database

✓ [Click here for details](#)

"Sanity" tests are tested that are expected to fail and have to be manually approved

The 'stdbuf -o0' is a trick to prevent the pipe from buffering the output, since in addition to storing it we also want to examine the output visually.

compara_external_foreign_keys

```
time bsub -M8000 -R"select[mem>8000] rusage[mem=8000]" -I stdbuf
-o0 ./run-configurable-testrunner.sh $(mysql-ens-compara-prod-1
details script) -d ensembl_compara_89 --release 89
--compara_master.database ensembl_compara_master -g ComparaSanity |
tee healthchecks_sanity_after_merge.txt
```

Typically, there are three failures:

- CheckConservationScoreSanity: Most of the multiple-genome WGAs will have a few blocks with no conservation scores. This is fine as long as it's only a few of them.
- CheckSyntenySanity: For distant species, some chromosomes may not have any synteny blocks. Again, there shouldn't be more than a few.
- CheckTableSizes: This compares the size of each table to the previous versions of the database and complains if there is a significant change (either way), if the tables have exactly the same size, or if some tables have appeared / disappeared. As the relco, you need to check that all the differences correspond to schema or dataset changes

Test web server

✓ Ask ensembl-production to point the test web server to the compara release database

Upon confirmation from the release coordinator ask other members of Compara to check their data on:
<http://staging.ensembl.org/>

Final handover of databases [\[edit\]](#)

Handover

- ☒ Let the production team know that our database is ready, and where. From this point onwards, everybody **hands-off** the database. Nothing is allowed to change in it
- ☒ If we have generated a new "Age of Base" file, it has to be copied to `/nfs/ensnfs-dev/staging/homo_sapiens/GRCh38/compara/` (usually it's named `Hsap_ages_${mlss_id}_${release_number}.bb`).
Let the web-team know if you have copied a new file or if they should consider the file from the previous release .
- ☒ Update the Declaration of Intentions on the admin website to indicate what has been handed over and what didn't make it and has been postponed

Final bits

- ☒ Dump the master database and place the copy in a safe place
 [Click here to expand...](#)

It should take a couple of minutes at most to run:

dump master database

```
become -- compara_ensembl
db_cmd.pl $COMPARA_REG compara_master --executable mysqldump | gzip
- >
/nfs/production/panda/ensembl/warehouse/compara/master_db_dumps/ensembl_compara_master_${CURR_ENSEMBL_RELEASE}.mysql.gz
```

Post-handover

Update documentation and diagrams [\[edit\]](#)

It is now time to update the static files. This should be done before we branch the code

- ☒ Update the pipeline diagrams for all the pipelines that have been run this release
 [Click here for details](#)
 Go to the docs directory

pipeline diagrams

```
cd $ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/production/diagrams
generate_graph.pl $COMPARA_REG compara_ptrees -output
ProteinTrees.png
generate_graph.pl $COMPARA_REG compara_nctrees -output
ncRNATrees.png
generate_graph.pl $COMPARA_REG compara_families -output
Families.png
generate_graph.pl --url
mysql://ensadmin:ensembl@compara5/mp14_dump_trees_protein_default_8
7 -out DumpTrees.png
generate_graph.pl --url
mysql://ensadmin:ensembl@compara1/mp14_dump_multi_align_human_vs_mo
use_87 -out DumpMultiAlign.png
generate_graph.pl -url mysql://ensro@compara3/kb3_pecan_20way_71
-output MercatorPecan.png
generate_graph.pl -url mysql://ensro@compara4/sf5_epo_35way_68
-output EpoLowCoverage.png
generate_graph.pl -url
mysql://ensro@compara4/sf5_compara_epo_13way_69 -output epo_pt3.png
```

Commit any changed diagrams to git and push.

- ☒ Update the schema documentation and diagrams
 ▼ [Click here for details](#)

generate new schema documentation

```
perl $ENSEMBL_CVS_ROOT_DIR/ensembl-production/scripts/sql2html.pl
-i $ENSEMBL_CVS_ROOT_DIR/ensembl-compara/sql/table.sql -o
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/schema/compara_schema.ht
ml -d Compara $(mysql-ens-compara-prod-1 details script) -dbname
ensembl_compara_88 -sort_headers 0 -sort_tables 0 -intro
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/schema/intro.html
```

Open the output file `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/schema/compara_schema.html` in your browser and check that no example errors are reported.

If everything looks fine, copy this file to public-plugins and commit&push both (the compara one and the webcode one) :

update schema documentation for web

```
cp
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/schema/compara_schema.ht
ml
$ENSEMBL_CVS_ROOT_DIR/public-plugins/docs/htdocs/info/docs/api/comp
ara/
```

If necessary, update schema diagrams by loading the `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/sql/table.sql` schema file into MySQL Workbench, rearrange/colour the nodes and export into PNG.

The schema diagrams will have to be copied both to `$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/schema/diagrams` and to

public-plugins and committed&pushed in both repositories:

update schema diagrams for web

```
cp -r
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/schema/diagrams/*.png
$ENSEMBL_CVS_ROOT_DIR/public-plugins/docs/htdocs/info/docs/api/compara/diagrams/
```

- ☒ Update the API tutorial documentation

▼ [Click here for details](#)

Update the tutorial documentation compara_tutorial.html in this directory:

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-webcode/htdocs/info/docs/api/compara/
```

Be careful that the embedded Perl snippets must use HTML-escaped characters (e.g. < and >) and be wrapped in a <pre class="code sh_perl">

Open the URL /info/docs/api/compara/compara_tutorial.html from a sandbox / test website and export it as a PDF in

```
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/ComparaTutorial.pdf
```

To make the pdf look nicer, you can issue a few JavaScript commands to remove the Ensembl headers. See [Creating PDF version of VEP docs](#) for more details

- ☒ Update the tutorial about Compara resources

▼ [Click here to expand...](#)

Do the same with \$ENSEMBL_CVS_ROOT_DIR/ensembl-webcode/htdocs/info/website/tutorials/compara.html . This pages can be viewed online at <http://staging.ensembl.org/info/website/tutorials/compara.html>

- ☐ Check examples work in ensembl-compara/scripts/examples/

- ☐ Create a word document and a pdf dump of this document

▼ [Click here for details](#)

In the top-right menu of this Confluence page, choose "Tools -> Export to PDF" and "Tools -> Export to Word".

Put these files into \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/production

- ☐ git commit and push any modified files or added tutorial examples

- ☐ **THIS IS NOW DONE EARLIER IN THE RELEASE PROCESS BEFORE THE DECLARATION OF INTENTION IS RELEASED TO THE PUBLIC!!! Update the declared intention with removed / deprecated methods

▼ [Click for details](#)

We need to generate the list of methods exported by the objects / adaptors on the master and release/{n-1} branches, and compare (diff) them.

Check deprecated / removed methods

```
# on master/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " *pm | sort > ~/MASTER_objects
# on the previous
release/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " *pm | sort > ~/RELEASE87_objects
sdiff -w 200 -bs ~/RELEASE87_objects ~/MASTER_objects | less

# Let's do the same for the adaptors
# on master/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " DBSQL/*pm | sort > ~/MASTER_adaptors
# on the previous
release/ensembl-compara/modules/Bio/Ensembl/Compara/
grep "^sub " DBSQL/*pm | sort > ~/RELEASE87_adaptors
sdiff -w 200 -bs ~/RELEASE86_adaptors ~/MASTER_adaptors | less
```

In both cases, make sure the methods are really removed, and not moved to a base / sub class, etc

Patch the test databases

- ☒ Be sure you are on a fresh master and follow the instructions from Patching test databases
Check that all the databases have been patched with

```
grep "schema_version." modules/t/test-genome-DBs/*/*/meta.txt
```

Rerun the entire test-suite to check that it still passes (fix it otherwise !)

```
prove -rv modules/t/
```

Commit and push all the changes under modules/t/test-genome-DBs/

Branch the code

- ☐ Check with the rest of Compara that it is ok to branch the code as it is, then create the 'release/THIS_RELEASE_NUMBER' branch in git and push it to the server.

```
cd $ENSEMBL_CVS_ROOT_DIR/ensembl-compara
git pull master
git branch release/<release number>
git checkout release/<release number>
git push origin release/<release number>
```

Specialize the branch

- ☐ The name of the branch ("master") is hardcoded in a number of places. Because the release branch is meant to be coupled to the other release branches, replace "master" with "release/<release number>" in these places
 - README.md . Replace all occurrences of "master" **except** the `git checkout master` in the installation guide of progressive-Cactus
 - .travis.yml . For all the repos that are eventually branched (ensembl-XXX, except ensembl-hive, and the external dependencies) we want to switch over to release/XX, but since they may not yet be branched, we also need to add a fallback to master, e.g.

```
# replace 'git clone --branch master --depth 1
https://github.com/Ensembl/ensembl-rest.git' with:
git clone --branch release/89 --depth 1
https://github.com/Ensembl/ensembl-rest.git || git clone --branch
master --depth 1 https://github.com/Ensembl/ensembl-rest.git
```

Take a look at .travis.yml on the release/89 branch for an example

However on master, we want to stick to the master branch of all dependencies, so here are the commands to update the files on the release/XX branch only without affecting master

```
cd $ENSEMBL_CVS_ROOT_DIR/ensembl-compara
git checkout release/<release number>
(edit and commit the files as explained above)
git push origin release/<release number>
git checkout master
git merge -s ours release/<release number>
git diff origin/master..master # should be empty
git push origin master
```

Pay attention to the merge command. The "-s ours" option tells git to make a merge commit that doesn't change any files (git diff will confirm that). The merge is still important so that bugfixes put on the release branch can be naturally merged with a standard git merge.

Test the sites

Once the branch is in, the web and core teams will set up the test sites. Each one has to be tested ideally following the same protocol as before the handover

- ☐ Main ensembl site <http://test.ensembl.org>
- ☐ GRCh37 site: <http://test.ensembl.org>
- ☐ Main REST server: <http://test.rest.ensembl.org>
Open each Compara endpoint and check that there is an output and that it is similar to the live site
- ☐ GRCh37 REST server: <http://test.grch37.rest.ensembl.org/>

Data dumps

The WGA and GERP dump pipelines require access to the core databases, and can only start once the Production team has given the go-ahead for this. This is to avoid overloading the databases whilst biomaRt is being run.

Most dumps (except homology) are currently generated in `/nfs/production/panda/ensembl/compara/${USER}/dumps_XX`, areas and then have to be manually assembled into the `/nfs/production/panda/ensembl/production/ensemblftp/release-XX/` tree.

Look at the previous release tree to get the idea. The first level of directories normally defines the file type, and the second level is the team name

(except fasta/ where species are mixed).

Ensembl-compara is responsible for the following dumps:

- bed/ensembl-compara (MSA)
- emf/ensembl-compara (MSA and homologies)
- maf/ensembl-compara (multiple_alignments and pairwise_alignments)
- tsv/ensembl-compara (homologies)
- xml/ensembl-compara (homologies)
- fasta/ancestral_alleles (the only one without ensembl-compara in the path)
- compara/

Make sure they are all either generated by running new dumps, or copied over from the previous release (if some MSAs did not run in this release /nfs/production/panda/ensembl/production/ensemblftp/release-XX/ tree.

All the copying has to be done with the compara_ensembl virtual user

Refer to the Intentions page for a list of what has to be run. Example databases are also listed under [List of all production pipelines](#)

☐ DNA data dumps

The instructions are in \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/docs/production/READMEs/multi_align.dumps.txt

☒ Gene tree dumps

▼ [Click here for details](#)

Go to \$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/modules/Bio/Ensembl/Compara/PipeConfig and open the PipeConfig file DumpTrees_conf.pm

Check that you are happy about all parameters. In usual cases, they can all be set from the command line and the config file does not need editing.

Run init_pipeline.pl with the other pipeconfig, (which inherits from DumpTrees):

init_pipeline

```
init_pipeline.pl DumpAllTrees_conf.pm -dump_per_species_tsv 1 -host  
mysql-ens-compara-prod-1.ebi.ac.uk -port 4485
```

Then run the beekeeper.pl -loop command suggested by init_pipeline.pl . This pipeline took 2h8m to run in rel82.

It will produce protein_tree dumps in the directory pointed at by 'target_dir' parameter. ('/lustre/scratch110/ensembl/'. \$self->o('ENV', 'USER')). \$self->o('pipeline_name'))

Commit the DumpTrees_conf.pm file into git if you'd like to keep the changes.

Rsync the directory that has been created

☐ Copy the tree content dump for Uniprot

▼ [Click here for details](#)

The file 'target_dir/ensembl.GeneTree_content.{release}.txt.gz' needs to be copied to the EBI ftp server, and then MD5 checksum computed and stored next to it:

init_pipeline

```
cp
/nfs/production/panda/ensembl/compara/${USER}/dumps_XX/ensembl.Gene
Tree_content.e81.txt.gz
/nfs/ftp/pub/databases/ensembl/ensembl_compara/gene_trees_for_unipr
ot/
cd
/nfs/ftp/pub/databases/ensembl/ensembl_compara/gene_trees_for_unipr
ot
md5sum ensembl.GeneTree_content.e<CURR_RELEASE_NUMBER>.txt.gz >
ensembl.GeneTree_content.e<CURR_RELEASE_NUMBER>.txt.gz.MD5SUM
```

☐ species trees

▼ [Click here for details](#)

This are the files found in /nfs/production/panda/ensembl/production/ensemblftp/release-XX/compara/species_trees: They can be copied from the previous release but some will need to be generated anew depending what production db we run in this release.

init_pipeline

```
#use the `DumpSpeciesTrees_conf` pipeline to dump the trees for the
production dbs.
#run the dumps as yourself but copy the file to the right location
afterwards as "compara_ensembl" user.
```

```
init_pipeline.pl
Bio::Ensembl::Compara::PipeConfig::DumpSpeciesTrees_conf
-compara_url
mysql://ensro@mysql-treefam-prod.ebi.ac.uk:4401/mateus_protein_trees_89
#remember to copy the new species trees to
/nfs/production/panda/ensembl/production/ensemblftp/release-XX/compara/species_trees
before running the pipeline again on another
production db
init_pipeline.pl
Bio::Ensembl::Compara::PipeConfig::DumpSpeciesTrees_conf
-compara_url
mysql://ensro@mysql-ens-compara-prod-3:4523/muffato_ensembl_ebinc_rna_trees_89c
```

```
init_pipeline.pl
Bio::Ensembl::Compara::PipeConfig::DumpSpeciesTrees_conf
-compara_url
mysql://ensro@mysql-ens-compara-prod-1:4485/muffato_mouse_strain_homologies_89
```

```
#to dump individual trees i.e ensembl_species-tree_NCBI_Taxonomy.nh
and ensembl_species-tree_Ensembl.nh use
ensembl-compara/scripts/example/species_getSpeciesTree.pl
perl species_getSpeciesTree.pl -url
'mysql://ensadmin:ensembl@mysql-ens-compara-prod-1.ebi.ac.uk:4485/ensembl_compara_89'
-mlss_id 60004 -label 'Ensembl'
#this will print the tree to the screen. copy and paste the tree
into the right file e.g ensembl_species-tree_Ensembl.nh
perl species_getSpeciesTree.pl -url
'mysql://ensadmin:ensembl@mysql-ens-compara-prod-1.ebi.ac.uk:4485/ensembl_compara_89'
-mlss_id 60004 -label 'NCBI Taxonomy'
#same as above
```

Ancestral alleles (for the Variation team)



Click here to expand...

Ancestral alleles are computed from EPO data. These can be symlinked from /warehouse/ens_ftp_arch_03/release-XX if no new alignments have been run

Repeat the process for each species of interest and perform a checksum when all data is tarred

```
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/ancestral_sequences/g
et_ancestral_sequence.pl --conf
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/pipeline/production_r
eg_ebi_conf.pl --compara_url
mysql://ensro@compara5/sf5_epo_8primates_77 --species homo_sapiens
dirname=homo_sapiens*
cd $dirname
perl
$ENSEMBL_CVS_ROOT_DIR/ensembl-compara/scripts/ancestral_sequences/g
et_stats.pl > summary.txt
cd ..
tar -cvzf ${dirname}.tar.gz $dirname
md5sum *.tar.gz > MD5SUM # when archives for all required species
are complete
```

☐ Let the production team know that the dumps are ready in their common location.