

Configuration

- These instructions show how to configure a new S-Match for later use with CHAIn. Currently the configuration sets up the new version to be used with SPSM. From there it is simple to also use the new version with CHAIn (see below under *Running CHAIn with new version*).
- Requirements: Working CHAIn installation
- Download newest S-Match version from: <http://sourceforge.net/projects/s-match/files/>
- Copy new S-Match version in /spsm/ folder
- Copy prolog-spsm.sh and all-spsm.sh from s-match-yyyyymmdd(old)/bin/ to s-match-yyyyymmdd(new)/bin/
- Change permission to allow executing files as a program for:
 - s-match-yyyyymmdd(new)/all-spsm.sh
 - s-match-yyyyymmdd(new)/prolog-spsm.sh
 - s-match-yyyyymmdd(new)/match-manager.sh
- Create s-match-spsm.properties file in s-match-yyyyymmdd(new)/conf:
 - (delete current s-match-spsm.properties file)
 - copy s-match.properties, rename it to s-match.properties and replace the TreeMatcher

Find these lines:

```
TreeMatcher=it.unitn.disi.smatch.matchers.structure.tree.spsm.SPSMTreeMatcher
TreeMatcher.SPSMTreeMatcher.mappingFactory=Global.MappingFactory
TreeMatcher.SPSMTreeMatcher.nodeMatcher=it.unitn.disi.smatch.matchers.structure.node.DefaultNodeMatcher
TreeMatcher.SPSMTreeMatcher.nodeMatcher.DefaultNodeMatcher.SATSolver=it.unitn.disi.smatch.deciders.CachingSolver
TreeMatcher.SPSMTreeMatcher.nodeMatcher.DefaultNodeMatcher.SATSolver.CachingSolver.SATSolver=it.unitn.disi.smatch.deciders.Min
iSAT
TreeMatcher.SPSMTreeMatcher.spsmFilter=it.unitn.disi.smatch.filters.SPSMMappingFilter
TreeMatcher.SPSMTreeMatcher.spsmFilter.SPSMMappingFilter.mappingFactory=Global.MappingFactory
```

And replace them with:

```
TreeMatcher=it.unitn.disi.smatch.matchers.structure.tree.spsm.SPSMTreeMatcher
TreeMatcher.SPSMTreeMatcher.mappingFactory=Global.MappingFactory
TreeMatcher.SPSMTreeMatcher.nodeMatcher=it.unitn.disi.smatch.matchers.structure.node.DefaultNodeMatcher
TreeMatcher.SPSMTreeMatcher.nodeMatcher.DefaultNodeMatcher.SATSolver=it.unitn.disi.smatch.deciders.CachingSolver
TreeMatcher.SPSMTreeMatcher.nodeMatcher.DefaultNodeMatcher.SATSolver.CachingSolver.SATSolver=it.unitn.disi.smatch
h.deciders.MinisAT
TreeMatcher.SPSMTreeMatcher.spsmFilter=it.unitn.disi.smatch.filters.SPSMMappingFilter
TreeMatcher.SPSMTreeMatcher.spsmFilter.SPSMMappingFilter.mappingFactory=Global.MappingFactory
```

- Create s-match-spsm-prolog.properties file in s-match-yyyyymmdd(new)/conf:
 - copy s-match-spsm.properties and rename it to s-match-spsm-prolog.properties
 - Change PlainMappingRenderer in
MappingRenderer=it.unitn.disi.smatch.renderers.mapping.**PlainMappingRenderer**
to **PrologMappingRenderer**
- Copy PrologMappingRenderer.java from s-match-yyyyymmdd(old)/src/it/unitn/disi/smatch/renderers/mapping to s-match-yyyyymmdd(new)/src/it/unitn/disi/smatch/renderers/mapping
- Rebuild java project:
 - go to folder s-match-yyyyymmdd(new)

- issue command `ant jar`
- Create `prolog-spsm-yyyy.sh` (for separately executing and testing the new S-Match version):
 - go to `/spsm/` folder
 - copy `prolog-spsm.sh` and rename it to `prolog-spsm-yyyy.sh`
 - Change `export S_MATCH_HOME` to `$DAISY_HOME_LOCAL/spsm/s-match-20yymmdd` (basically folder in which new S-Match version is residing)
 - Change
 - `./prolog-spsm.sh $SPSM_HOME/spsm-match-data/source.txt $SPSM_HOME/spsm-match-data/target.txt $SPSM_HOME/spsm-match-data/result.txt > /dev/null`
 - to `./prolog-spsm.sh $SPSM_HOME/spsm-match-data/source.txt $SPSM_HOME/spsm-match-data/target.txt $SPSM_HOME/spsm-match-data/result-yyyy.txt > /dev/null`

First run

- Run both `prolog-spsm.sh` and `prolog-spsm-yyyy.sh`
- Go to folder `/spsm-match-data/`
- Compare `result.txt` and `result-yyyy.txt`
- The files should show the same results
- If the results are not the same, then it is necessary to investigate whether this is caused by an incorrect configuration of properties in `s-match-yyyyymmdd(new)/conf` or by updates made to the S-Match core or SPSM, respectively (the former needs to be fixed, while the latter two might justify obtaining a different result)

Starting points for investigating potential differences between old and new S-Match versions

- Use other existing property files in `match-yyyyymmdd(old)/bin/` to `s-match-yyyyymmdd(new)/bin/prolog-spsm.sh`
 - Change `s-match-spsm-prolog.properties` to compare results and determine where the difference does not originate from
 - Change `./match-manager.sh` online `$1.xml $2.xml $3 -config=../conf/s-match-spsm-prolog.properties -property=ContextLoader=it.unitn.disi.smatch.loaders.context.SimpleXMLContextLoader`
 - to `s-match.properties` to only run the query with S-Match (without `PrologMappingRenderer` and without `tree-edit-distance mapper`)
 - to `s-match-spsm.properties` to run the query with SPSM and S-Match (without `PrologMappingRenderer`, but with `tree-edit-distance mapper`)
- Run `all-cw.sh` and `all-spsm.sh` in `s-match-yyyyymmdd(old)/bin/` and `s-match-yyyyymmdd(new)/bin/` and compare results
- Consult release notes, if available
- Compare files of new and old S-Match versions, e.g. through
 - `diff -qr s-match-yyyyymmdd(new) s-match-yyyyymmdd(old) | sort`
 - by installing `meld`: `apt-get install meld`

Running CHAIIn with new version

- The easiest way to run CHAIIn with the new version is to edit /spsm/prolog-spsm.sh
 - Change export S_MATCH_HOME=\$DAISY_HOME_LOCAL/spsm/s-match-yyyymmdd(old)
 - to export S_MATCH_HOME=\$DAISY_HOME_LOCAL/spsm/s-match-yyyymmdd(new)
- Instead of changing spsm/prolog-spsm.sh it might be prudent to create a copy of /queryRespond/translation.sh that uses the new version
- Ideally translation.sh would be best set up to receive an argument indicating which S-Match version to use