IMMERSE "share a config"

Planned scheme how to
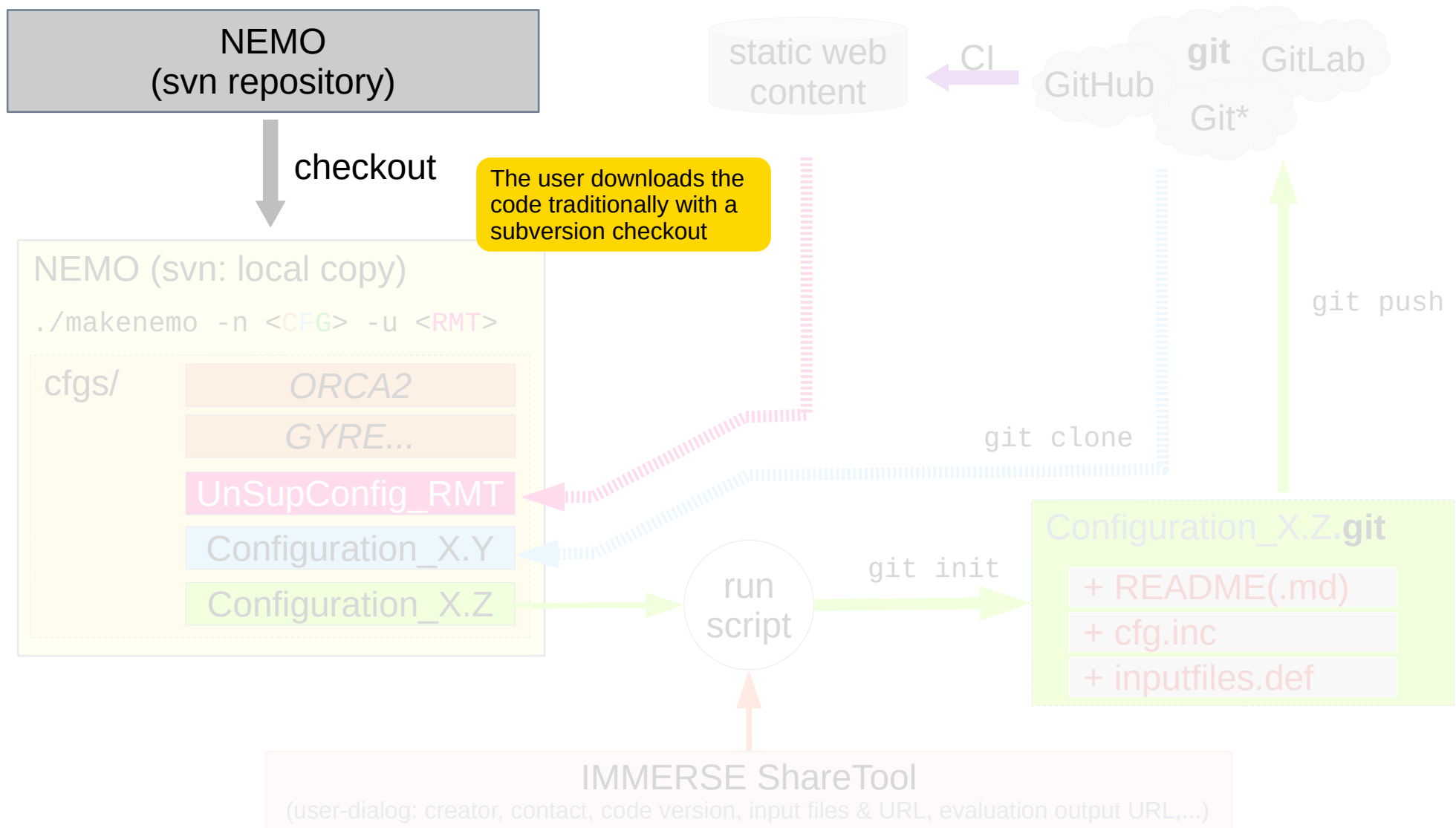## provide[1] or share[2] a non-ref[3] NEMO configuration[4]

Markus Scheinert (mscheinert@geomar.de)
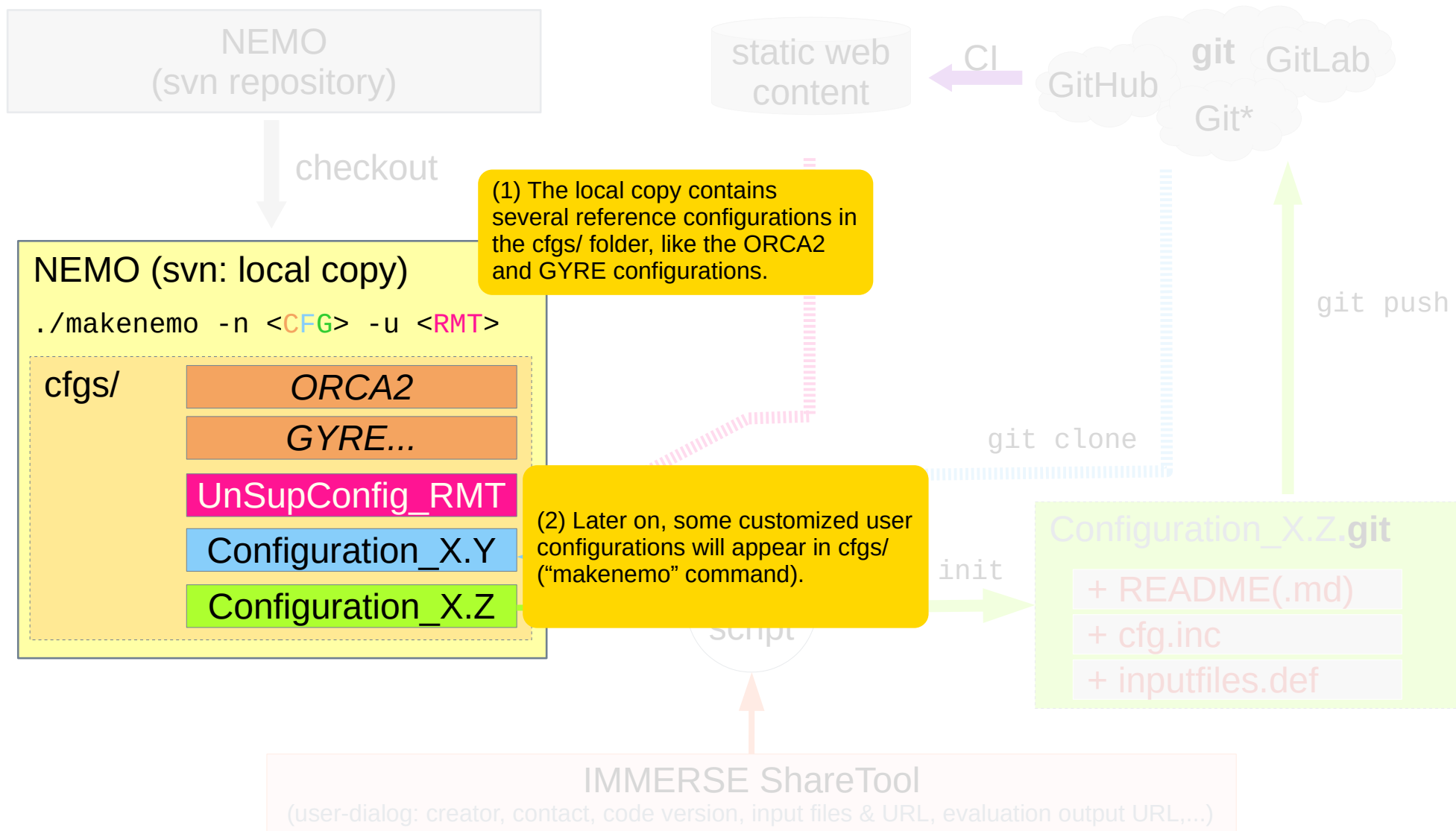GEOMAR, Kiel | 2019-06-25

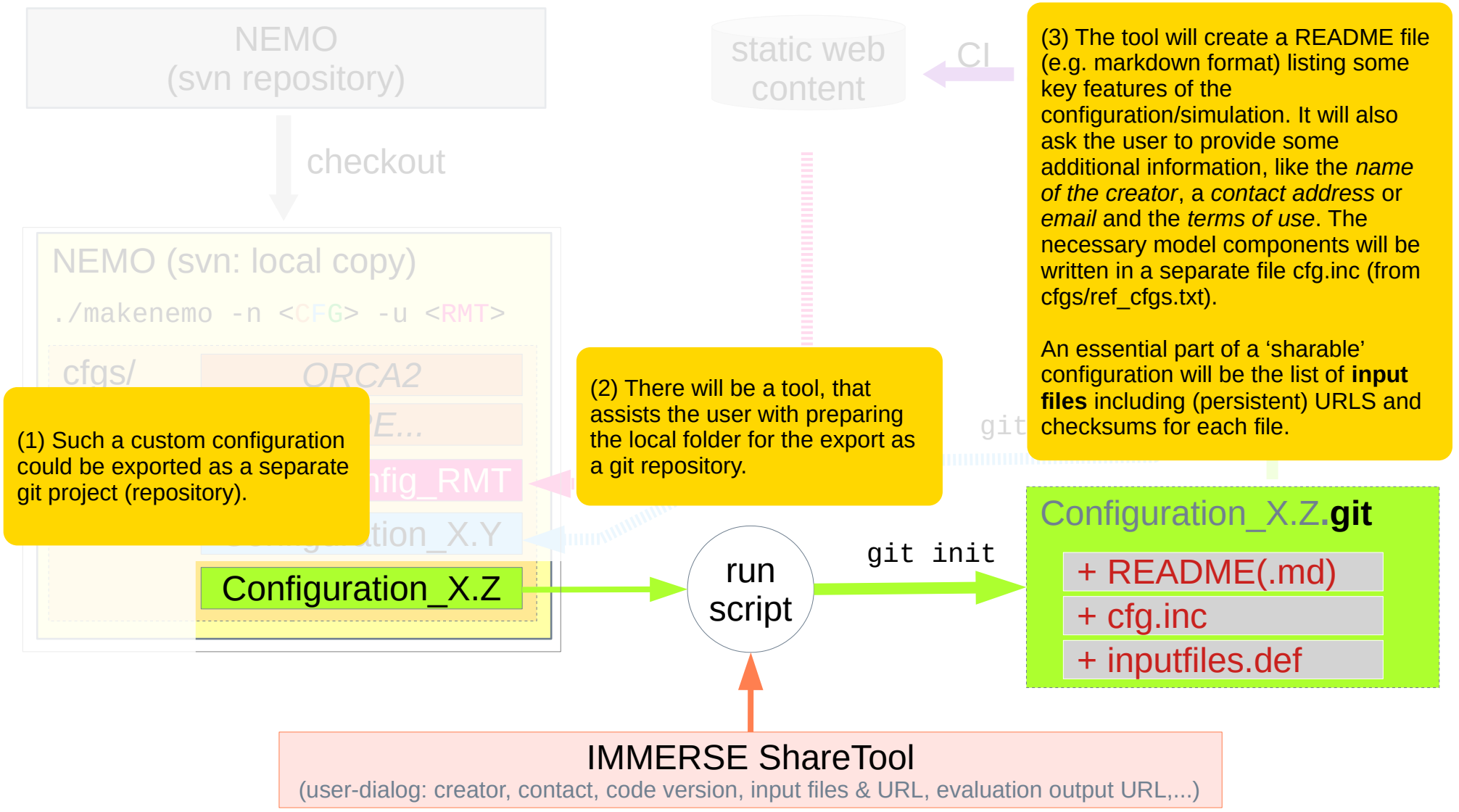[1] Provide the details of a configuration to other users
[2] Collaborative development of a configuration
[3] A configuration that is not shipped with the NEMO source code package
[4] A "sharable configuration" always consists of: definite **reference to the code base**, **modified code**, list of model **components** and **cpp-keys** as well as **namelists** and reliable URLs to **input files** (incl. a checksum)

NEMO
(svn repository)

checkout

static web
content

CI

NEMO (svn: local copy)

./makenemo -n <CFG> -u <RMT>

cfgs/   ORCA2
                         E...

config_RMT

tion_X.Y

Configuration_X.Z

git

(3) The tool will create a README file (e.g. markdown format) listing some key features of the configuration/simulation. It will also ask the user to provide some additional information, like the *name of the creator*, a *contact address* or *email* and the *terms of use*. The necessary model components will be written in a separate file cfg.inc (from cfgs/ref_cfgs.txt).

An essential part of a 'sharable' configuration will be the list of **input files** including (persistent) URLS and checksums for each file.

(2) There will be a tool, that assists the user with preparing the local folder for the export as a git repository.

(1) Such a custom configuration could be exported as a separate git project (repository).

run
script

git init

Configuration_X.Z.**git**

+ README(.md)

+ cfg.inc

+ inputfiles.def

IMMERSE ShareTool
(user-dialog: creator, contact, code version, input files & URL, evaluation output URL,...)

NEMO
(svn repository)

static web
content

CI

git GitLab

GitHub

Git*

checkout

NEMO (svn: local copy)

`./makenemo -n <CFG> -u <RMT>`

cfgs/

ORCA2

GYRE...

UnSupConfig_RMT

Configuration_X.Y

Configuration_X.Z

The git repository can then be exorted to a remote git server according to the user's choice.

`git push`

`git clone`

run
script

`git init`

Configuration_X.Z.**git**

+ README(.md)

+ cfg.inc

+ inputfiles.def

IMMERSE ShareTool

(user-dialog: creator, contact, code version, input files & URL, evaluation output URL,...)

NEMO
(svn repository)

checkout

static web   CI

git GitLab

GitHub

Git*

NEMO (svn: local copy)

./makenemo -n <CFG> -u <RMT>

Different groups/communities can share their configuration repositories in different groups/namespaces on different hosts.

Users with access permission can either download the code directly from the server (tar,zip,,..) or....

git push

cfgs/

ORCA2

GYRE...

UnSupConfig_RMT
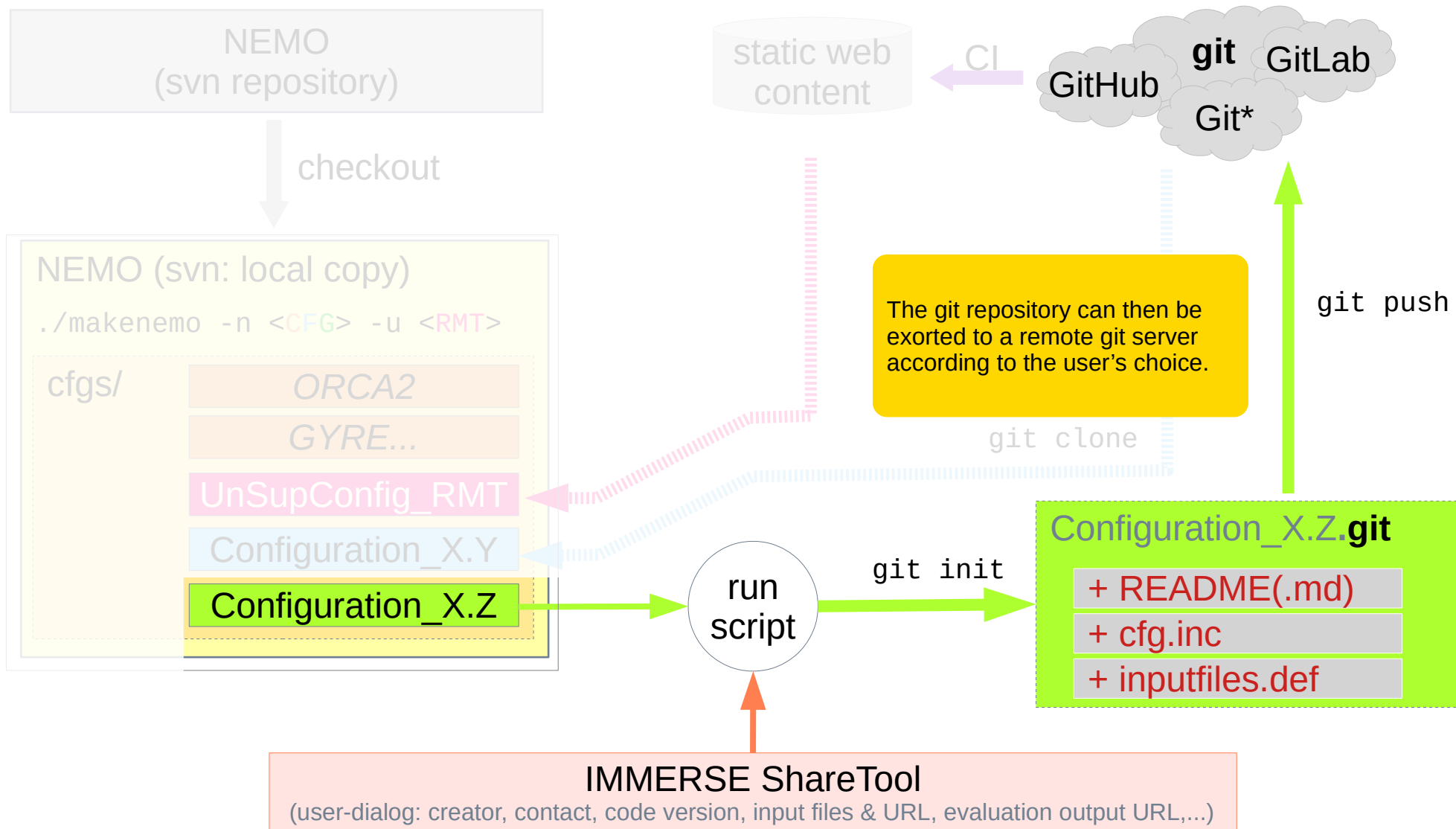
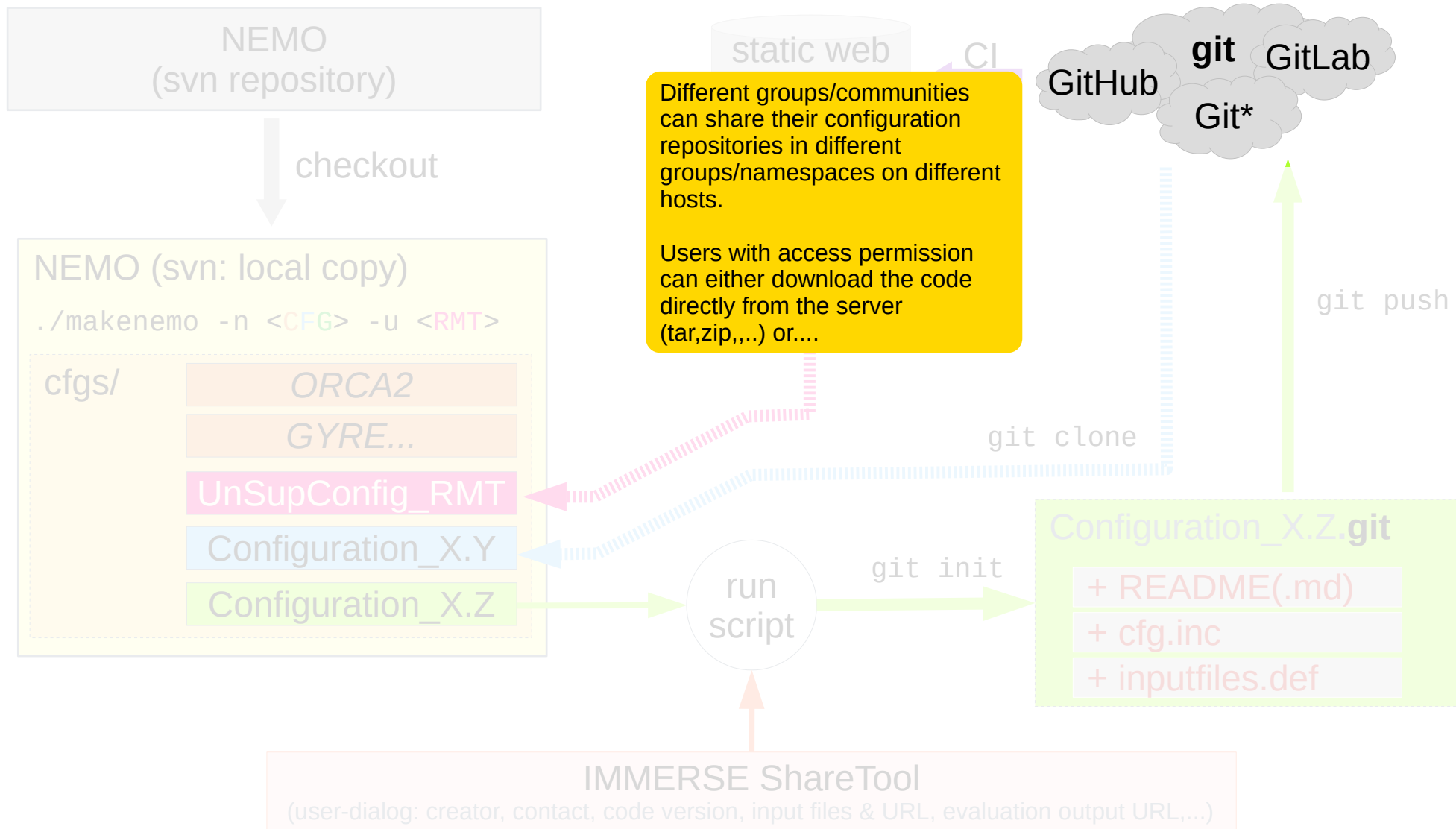Configuration_X.Y
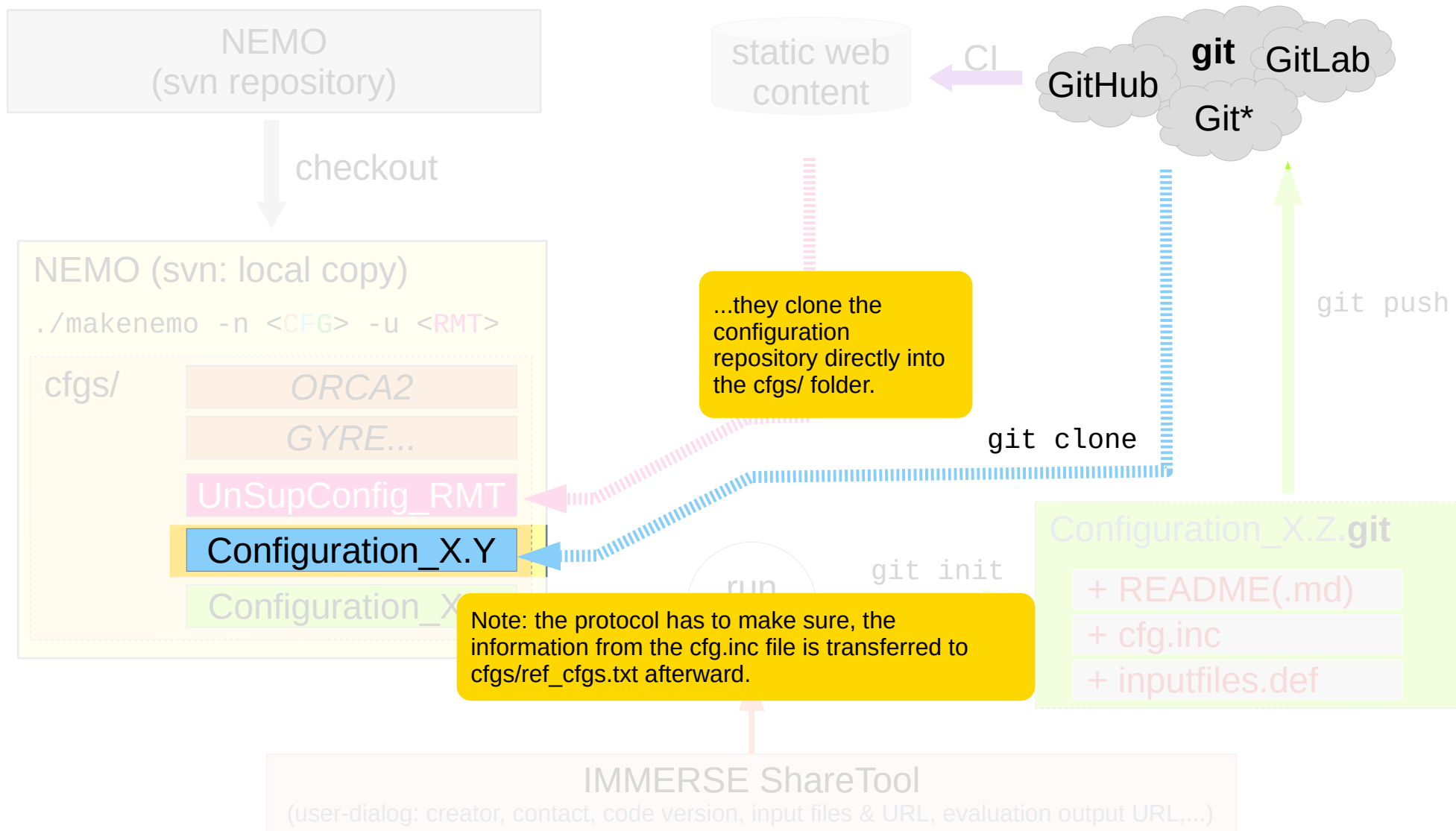
Configuration_X.Z

git clone

run
script

git init

Configuration_X.Z.**git**
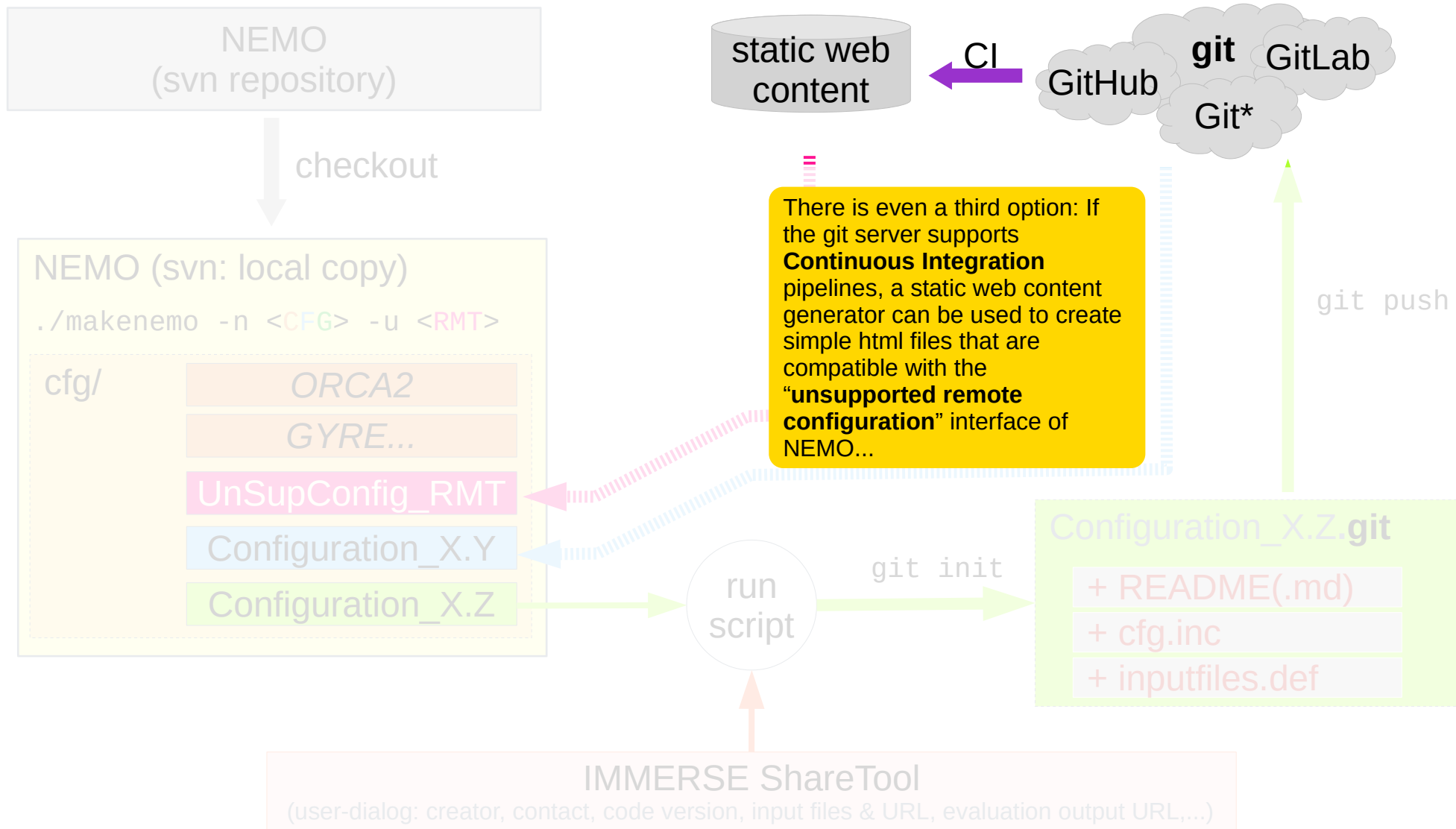
+ README(.md)

+ cfg.inc

+ inputfiles.def

IMMERSE ShareTool

(user-dialog: creator, contact, code version, input files & URL, evaluation output URL,...)

NEMO
(svn repository)

static web content

CI

git GitLab

GitHub

Git*

checkout

NEMO (svn: local copy)

./makenemo -n <CFG> -u <RMT>

cfgs/

ORCA2

GYRE...

UnSupConfig_RMT

Configuration_X.Y

Configuration_X...

...they clone the configuration repository directly into the cfgs/ folder.

git clone

git push

run

git init

Configuration_X.Z.**git**

+ README(.md)

+ cfg.inc

+ inputfiles.def

Note: the protocol has to make sure, the information from the cfg.inc file is transferred to cfgs/ref_cfgs.txt afterward.

IMMERSE ShareTool
(user-dialog: creator, contact, code version, input files & URL, evaluation output URL,...)

NEMO
(svn repository)

checkout

NEMO (svn: local copy)

`./makenemo -n <CFG> -u <RMT>`

cfg/

ORCA2

GYRE...

UnSupConfig_RMT

Configuration_X.Y

Configuration_X.Z

static web
content

CI

git GitLab

GitHub

Git*

There is even a third option: If
the git server supports
Continuous Integration
pipelines, a static web content
generator can be used to create
simple html files that are
compatible with the
"**unsupported remote
configuration**" interface of
NEMO...

git push

run
script

`git init`

Configuration_X.Z.**git**

+ README(.md)

+ cfg.inc

+ inputfiles.def

IMMERSE ShareTool
(user-dialog: creator, contact, code version, input files & URL, evaluation output URL,...)

Unfortunately, the remote URL has to be registered first in the rmt_cfgs.txt file in the tests/ folder.

An enhancement of the makenemo script could be to allow an URL as an argument for the -u option. The model components then could be retrieved from the remote cfg.inc file.

static web content

CI

git GitLab

GitHub

Git*

git push

./makenemo -n <CFG> -u <RMT>

cfgs/ ORCA2

GYRE...

UnSupConfig_RMT

Configuration_X.Y

Configuration_X.Z

git clone

run script

git init

Configuration_X.Z.**git**

+ README(.md)

+ cfg.inc

+ inputfiles.def

IMMERSE ShareTool

(user-dialog: creator, contact, code version, input files & URL, evaluation output URL,...)