

Migrating to Rstudio Server

Experiences from Statistics Austria

Bernhard Meindl (bernhard.meindl@statistik.gv.at)
Alexander Kowarik (alexander.kowarik@statistik.gv.at)

The Hague, September 2018

What's the plan?

Sharing the experiences at Statistics Austria we made while switching the R-Users from Rstudio Desktop → Rstudio Server

What's the plan?

Sharing the experiences at Statistics Austria we made while switching the R-Users from Rstudio Desktop → Rstudio Server

Reasons:

Why did we decide to switch?

What's the plan?

Sharing the experiences at Statistics Austria we made while switching the R-Users from Rstudio Desktop → Rstudio Server

Reasons:

Why did we decide to switch?

Timeline

When did we switch and how long does it take?

What's the plan?

Sharing the experiences at Statistics Austria we made while switching the R-Users from Rstudio Desktop → Rstudio Server

Reasons:

Why did we decide to switch?

Timeline

When did we switch and how long does it take?

Setup/Migration and Status Quo

What did we do? How far are we and what problems do we still have?

What's the plan?

Sharing the experiences at Statistics Austria we made while switching the R-Users from Rstudio Desktop → Rstudio Server

Reasons:

Why did we decide to switch?

Timeline

When did we switch and how long does it take?

Setup/Migration and Status Quo

What did we do? How far are we and what problems do we still have?

Lessons learned

Reasons

Why did we want to switch?

Some History

Starting point

- Methods division prepared **R** installation package consisting of
 - **R** + **RStudio** + LaTeX + development tools (compilers, git, ...)
- Rollout by IT division for all (~100) **R** installations

Some History

Starting point

- Methods division prepared **R** installation package consisting of
 - **R** + **RStudio** + LaTeX + development tools (compilers, git, ...)
- Rollout by IT division for all (~100) **R** installations

Problem

- difficult to maintain (eg. when to update) and to keep up-to date

Some History

Starting point

- Methods division prepared **R** installation package consisting of
 - **R** + **RStudio** + LaTeX + development tools (compilers, git, ...)
- Rollout by IT division for all (~100) **R** installations

Problem

- difficult to maintain (eg. when to update) and to keep up-to date

But

- Methods division always had access to a Linuxserver running Rstudio-Server (Open-Source version)
- Our experience in using the server was great!
- → Idea: create environment so that all **R**-users at STAT have access to the server version of Rstudio

Timeline (1)

2017: Preparation

- Autumn: Decision was made to migrate all **R**-users to Rstudio Server
- Cooperation with IT-department (important):
 - Cast of a deployment strategy and specific tasks
 - Setup of two (identical) Linux servers (test and production)
 - Setup of Rstudio-Server Pro (with evaluation license)

Timeline (1)

2017: Preparation

- Autumn: Decision was made to migrate all **R**-users to Rstudio Server
- Cooperation with IT-department (important):
 - Cast of a deployment strategy and specific tasks
 - Setup of two (identical) Linux servers (test and production)
 - Setup of Rstudio-Server Pro (with evaluation license)

2017: Evaluation and Testing

- Testing and Evaluation of Rstudio-Server with very positive results
- Useful features include:
 - per user/group resource allocation
 - multiple **R** versions
 - multiple concurrent sessions

Timeline (2)

2018: Training and Adjustments

- Design of a training course for the new setup
- Rewrite of existing **R**-courses to refer to the new infrastructure
- Transfer of users from Desktop to Server
- But: Still maintaining the latest **R** desktop installation as "*fallback*"
- Testing and Installation of Rstudio-Connect (easy deployment of Shiny apps, APIs and reproducible reports)

Timeline (2)

2018: Training and Adjustments

- Design of a training course for the new setup
- Rewrite of existing **R**-courses to refer to the new infrastructure
- Transfer of users from Desktop to Server
- But: Still maintaining the latest **R** desktop installation as "*fallback*"
- Testing and Installation of Rstudio-Connect (easy deployment of Shiny apps, APIs and reproducible reports)

2019: ??

- Removing the last Rstudio desktop installations? (perhaps :))

Setup (1)

Hardware Setup

- On both (virtualized) Test- and Productionserver we have:
 - Ubuntu 16.04, Xeon 8 Cores and 128GB RAM
 - easy to scale because it is virtualized
- Problem: difficult to come up with an estimate for peak-usage
- Our solution: estimate 4GB per user, consistently monitor usage/load and upgrade users on demand

Setup (1)

Hardware Setup

- On both (virtualized) Test- and Productionserver we have:
 - Ubuntu 16.04, Xeon 8 Cores and 128GB RAM
 - easy to scale because it is virtualized
- Problem: difficult to come up with an estimate for peak-usage
- Our solution: estimate 4GB per user, consistently monitor usage/load and upgrade users on demand

Software

- Rstudio-Server, Shiny-Server and Rstudio-Connect are installed
- we can also make use of internal Bitbucket installation (git)

Setup (1)

Hardware Setup

- On both (virtualized) Test- and Productionserver we have:
 - Ubuntu 16.04, Xeon 8 Cores and 128GB RAM
 - easy to scale because it is virtualized
- Problem: difficult to come up with an estimate for peak-usage
- Our solution: estimate 4GB per user, consistently monitor usage/load and upgrade users on demand

Software

- Rstudio-Server, Shiny-Server and Rstudio-Connect are installed
- we can also make use of internal Bitbucket installation (git)

Issues, Problems and Feedback

- Users get added to a specific Jira project where they can ask questions or file issues (but still: mail and phone popular)

Setup (2)

Why two servers?

- Testserver:
 - mostly used by staff of methods unit (us :))
 - we can test new versions of **R** (packages), **Rstudio-Server**, **Rstudio-Connect** and **Shiny-Server** before deploying on production server
- Production:
 - used by all other users
 - latest **R**-version usually is one minor-revision behind the Testserver

Setup (2)

Why two servers?

- Testserver:
 - mostly used by staff of methods unit (us :))
 - we can test new versions of **R** (packages), **Rstudio-Server**, **Rstudio-Connect** and **Shiny-Server** before deploying on production server
- Production:
 - used by all other users
 - latest **R**-version usually is one minor-revision behind the Testserver

Process

- well-defined process if a user wants to switch
 - IT departments creates user on production server
 - User gets authorization to use internal Bitbucket / Jira
 - User updates credentials (one-time ssh-login required)

Achievements

Where are we now?

- several training (migration) sessions held
- ~140 **R** users migrated to the server
- overall low number of problems and issues

Achievements

Where are we now?

- several training (migration) sessions held
- ~140 R users migrated to the server
- overall low number of problems and issues

Current work

- Fine-Tuning: helping users to get accustomed
- Rewriting training materials making use of the new infrastructure
- Identifying use-cases that "*block*" removal of RStudio-Desktop instances
- Writing some helper packages

Custom R Packages (1)

Why do we need internal R-packages?

- One major problem was to give users access to (windows-based) file shares
- **Reason:** No mapping between Windows- and Linux user credentials
- **Solution:** package mountSTAT

Custom R Packages (1)

Why do we need internal R-packages?

- One major problem was to give users access to (windows-based) file shares
- **Reason:** No mapping between Windows- and Linux user credentials
- **Solution:** package mountSTAT

mountSTAT

- allows to mount windows-based shares with *"real"* credentials
- secure (encrypted) way of saving credentials using package secret
- very easy usage

```
library(mountSTAT)
saveWindowsUser()
con <- mountWindowsShare(server="daten1",share="stat",mountp="d1")
list.files(con)
unmountWinShare(con)
```

Custom R Packages (2)

rinstSTAT

- this package allows to install a new **R** version (re-installing previously installed packages)
- it also allows to upgrade packages for specific **R** versions
- we can also use this package to update a local miniCRAN-repository containing development packages
- querying a custom API showing information on the R installation

```
library(rinstSTAT)
install_new_Rversion("3.5.1", old_library="/opt/R/3.5.0")
update_local_repository(newVersionOnly = FALSE)
deploy_devel_pkgs(update_all=TRUE)
stat_r_api(api="apiversion")
stat_r_api(api="rinfo", query="rvers=3.5.1")
stat_r_api(api="pkginfo", query="rvers=3.5.0&pkg=sdcmicro")
stat_r_api(api="versions")
stat_r_api(api="users")
```

Custom R Packages (3)

dataSTAT

- easy access to DB2 databases
 - contains drivers, hardcoded database names
 - performance improvements compared to RJDBC
- downloading files from/to sftp-servers
- read data from mainframe
- read files from ms-access-databases using the underlying `mdbtools` software

Custom R Packages (3)

dataSTAT

- easy access to DB2 databases
 - contains drivers, hardcoded database names
 - performance improvements compared to RJDBC
- downloading files from/to sftp-servers
- read data from mainframe
- read files from ms-access-databases using the underlying mdbtools software

and some more ...

- sampSTAT: creating, modifying, exporting internal sampling frames
- graphSTAT: interactively creating standardized graphs
- mzSTAT: methods to work with microcensus data

Experiences and Lessons learned

... that may be helpful for others

Experiences (1)

Take your time

- do not rush and make realistic assumptions on time required for the preparation, implementation and transition phase

Experiences (1)

Take your time

- do not rush and make realistic assumptions on time required for the preparation, implementation and transition phase

Cooperation with IT

- At some point you will need their help. Make them your friends!

Experiences (1)

Take your time

- do not rush and make realistic assumptions on time required for the preparation, implementation and transition phase

Cooperation with IT

- At some point you will need their help. Make them your friends!

Friendly users

- Identify some colleagues that are interested in trying out the new environment, providing useful inputs and "*free testing*"

Experiences (2)

Be aware of roadblocks

- Problems you might never have thought of will occur because people can (and do) crazy stuff with R

Experiences (2)

Be aware of roadblocks

- Problems you might never have thought of will occur because people can (and do) crazy stuff with R

Slow down

- do not try to migrate all users at once. Do it in "*batches*"

Experiences (2)

Be aware of roadblocks

- Problems you might never have thought of will occur because people can (and do) crazy stuff with R

Slow down

- do not try to migrate all users at once. Do it in "*batches*"

Training and Feedback

- allow (and reserve) extra time to prepare trainings and provide feedback to colleagues asking questions

Thanks for your attention

Any questions/comments?

Ask now or

Contact me: bernhard.meindl@statistik.gv.at