

## Community

[Ask a Question](#) [Write a Blog Post](#)

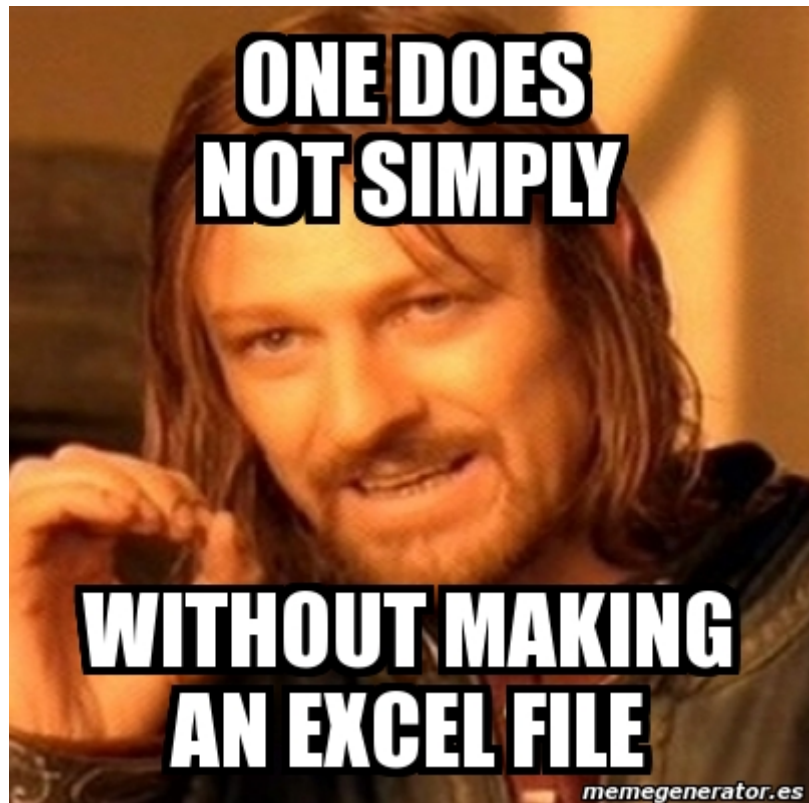
## Personal Insights



Lars Hvam

July 18, 2023 | 1 minute read

# ABAP & Excel, why can't we just have something nice?

[Follow](#)[0](#) [0](#) [1](#) [Edit](#)[Like](#)[RSS Feed](#)

Exporting data to Excel in ABAP is a [common topic](#), but there are many different options to choose,

- [abap2xlsx](#)
- CL\_FDT\_XL\_SPREADSHEET

- XCO\_CP\_XLSX
- CL\_SALV\_BS\_TT\_UTIL
- SAP\_CONVERT\_TO\_XLS\_FORMAT
- XXL\_FULL\_API
- and more

But, I just want one, one that works, one that is nice. But what is *nice*?

## 1. Works everywhere

Something that works everywhere, I don't want to learn different APIs for different platforms. Ideally this means 702 and up to latest [Steampunk](#).

Something like [abapGit](#) has worked for on-prem 702 and up for almost 10 years, so it is possible to build software that works on multiple versions.

abap2ui5 [does an automatic downport](#) in order to write the software in high syntax, while still supporting 702.

## 2. Public issue list

All software has bugs, some will be fixed, some not. But most important to me is easy access to check the issues, what are the known bugs, has someone had the same problem as me? It helps a lot to be able to search open and closed issues to find bugs in my code.

## 3. Suggest fixes

If I discovered a bug, I might have a fix for it, let me suggest that fix instead of having someone else spend the time to get to the same conclusion/fix.

## 4. Upgrade at any time

Let me upgrade the software at my convenience, not along with 1000s of other things. If I choose to consider it all working, there is no need to upgrade.

Reduce the impact by [having multiple copies](#) inside the same system, managed by different teams. That way each team can decide how/when to upgrade.

## 5. Easy testing

Easy ad-hoc testing, why require code to be installed? Something like [codesandbox](#), but with the ABAP code in the left editor. This exists, see [abap-openapi](#) as an example, it runs the ABAP code but in the browser, on the fly.

This also goes for unit tests, just edit a file in notepad and push it to github, and it will [trigger the unit tests running in seconds](#). See [ajson example](#)

## 6. Public versioning

If something breaks I want to be able to pull an old version, and or easily see the differences between the old and installed version. This makes it much easier to decide when to upgrade, or do debugging.

We keep getting more and more options, with fewer features, and a cluttered core plus ecosystem. The direction that SAP and the community is heading is just overhead for everyone.

Its the same story for JSON handling, HTTP clients and more in ABAP.

What's nice to you? And how do we get there?

[Alert Moderator](#)

### Assigned Tags

---

ABAP Development

#ABAP

#abap2xlsx

### Similar Blog Posts

---

[My biggest mistakes as an ABAP Developer](#)

By Mauricio Cruz Mar 19, 2015

[New ABAP Debugger Tipps and Tricks](#)

By Olga Dolinskaja Nov 09, 2010

[Hello, SAP, Where Are My Global Classes?](#)

By Jelena Perfiljeva May 24, 2022

### Related Questions

---

[Has The New SAP Community Killed The Community?](#)

By Former Member Oct 27, 2016

Where condition alternative for is not initial

By Former Member Jan 05, 2017

Watch ABAPers at SAP TechEd Getting Lemonade!

By Jason Lax Sep 19, 2017

Be the first to leave a comment

Add Comment

Find us on

Privacy	Terms of Use
Legal Disclosure	Copyright
Trademark	Cookie Preferences
Newsletter	Support