

# OSPOID: Case Title

---

- ID: OSPOID
- Requestors: !
- Stakeholders:
- OS-Supervisor: !
- Received:
- Version:
- Solved:

## 0.) Inhalt

---

- 1.) Description
  - 1.1) System/Product/Matter
  - 1.2) Architecture
  - 1.3) Used Open Source Software
- 2.) Use Case Analysis
- 3.) Compliance Activities
  - 3.1) The Project Specific *Don'ts*
  - 3.2) Licensing Your Code
  - 3.3) The Project Specific *Does*
- 4.) Conclusion

The step numbers in square brackets refer to the [OSPO get compliant products flowchart](#). That flowchart describes the sequential step-by-step process. The OSPO compliance case file (OCCF) you just read covers the process results systematically, not necessarily in accordance with the step-by-step sequence.

## 1. Description

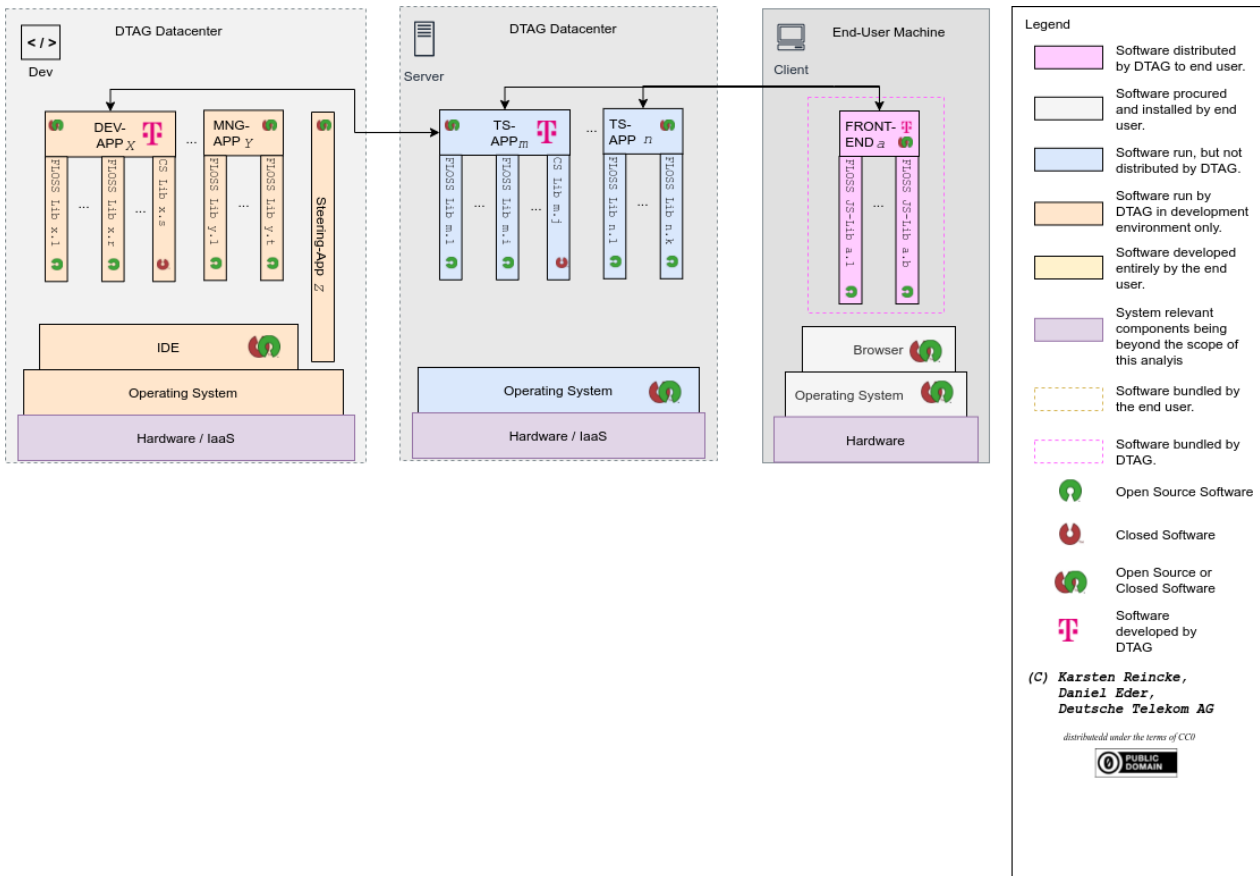
---

### 1.1 System/Product/Matter

Describe the purpose and features of the system you want to distribute / sell compliantly [→ S2.A]

### 1.2 Architecture

Describe the software architecture of the system you want to distribute / sell compliantly [→ S2.B]



### 1.3 Used Open Source Software

0) For each component of your system containing a self developed app create an **External Bill of Material** meaning a list of all its open source components that are assumed to be externally preinstalled by any third party [→ S5... particularly S5.X.b/d]

1) For each component of your system you want to distribute / sell compliantly (in accordance with the open source license) create the respective **Bill Of Material** meaning a list of all used open source applications and embedded open source libraries / modules / snippets [→ S5... particularly S5.X.c/e]

2) For each entry of each **BOM** extend the structure of the entry to that of a qualified BOM\*

- **Creating a qualified BOM** means For each entry of each **BOM** find and document the respective licensing statements by inserting into the respective line of the list
  - a link to the respective homepage
  - the respective release number (helpful)
  - a typing as app (executed in its own address stack and talking with the other app only via inter process communication) or lib (being included or linked to the overarching component)
  - a link to the source code repository
  - a link to the respective license file or licensing statement

#### 1.3.1 Development Tools

#### 1.3.2 Server

The externally preinstalled components

NO.	Component	Release	Type	Source Repo	License
-----	-----------	---------	------	-------------	---------

NO.	Component	Release	Type	Source Repo	License
1	COMP	??	App/lib	RepoUrl	SPDX
	[]()			[]()	[]()

The components bundled with the server

NO.	Component	Release	Type	Source Repo	License
1	COMP	??	App/lib	RepoUrl	SPDX
	[]()			[]()	[]()

### 1.3.3 Client / Frontend

The externally preinstalled components

NO.	Component	Release	Type	Source Repo	License
1	COMP	??	App/lib	RepoUrl	SPDX
	[]()			[]()	[]()

The components bundled with the distributed client

NO.	Component	Release	Type	Source Repo	License
1	COMP	??	App/lib	RepoUrl	SPDX
	[]()			[]()	[]()

## 1.4 Particularities

Keep in mind and comment (if necessary) the components licensed under the **AGPL-3.0** and libraries / modules licensed under any strong copyleft license (GPL-2.0, GPL-3.0)

## 2. Use Case Analysis

Thoroughly describe which of the components are - in any sense - handed over to any third party / customer [→ S2.C]

## 3. Compliance Activities

### 3.1 The Project Specific *Don'ts*

Come back to the OSPO which will discuss with you your specific list of DON'Ts [→ S2.D]

#### 3.1.1 Don'ts concerning the Development Tools

#### 3.1.2 Don'ts concerning the Server

#### 3.1.3 Don'ts concerning the Client / Frontend

## 3.2 Licensing Your Own Development

---

*Wherever you integrate a self-developed app into any component you must deal with potential strong copy left effects (either proof, that there isn't any strong copyleft effect triggered, or license you code under the same license) and consider how to license and distribute you self-developed app [→ S6...]*

### 3.2.1 Your App/Part in the Development Tools

...

### 3.2.2 Your App/Part in the Server

...

### 3.2.3 Your App/Part in the Client

...

## 3.3 The Project Specific *Does*

---

*You may use your open source components only if you create the required open source compliance artifacts and bundle and distribute them together with you system / product.*

*So, come back to the OSPO which will guide you to create these artifacts, namely the `__O__pen __S__ource __C__ompliance __F__ile` [→ S7...]*

### 3.3.1 Towards an OSDF for the Development Tools

### 3.3.2 Towards an OSDF for the Server

### 3.3.3 Towards an OSDF for the Client / Frontend

## 4. Conclusion

---

---



(C) 2021 Karsten Reincke, Deutsche Telekom AG: This file is distributed under the terms of the [CC0-license](#)

Deutsche Telekom makes no warranties about the work, and disclaims liability for all uses of the work, to the fullest extent permitted by applicable law.