Celonis

Operation Guide

Version 1.1

Corresponding Software Version: 4.0

CONTENTS

INTRODUCTION

ABOUT CELONIS

Celonis is a powerful software for retrieving, visualizing and analyzing real as-is business processes from transactional data. It provides users with the possibility to create and share comprehensive process analyses giving them full transparency about the business processes at hand.

In order to provide process analyses, Celonis makes use of raw data taken from your database system. For efficient use, the solution requires data to exist in a specified target structure. Thus, raw data from your system will be transformed into that structure on a regular basis. The result will then be stored as either views or tables.

Celonis was designed as an analysis platform for the supervision of several business processes and by several users at the same time. Therefore, Celonis is a browser-based web application with a client-server architecture that is easily accessible throughout an enterprise for many users at the same time. User access can be restricted either on analysis-, or data-model-level. Users can also be assigned different roles with different rights for making configurations or creating analyses.

TARGET AUDIENCE

This guide covers all relevant technical information about correctly operating Celonis and is meant to be consulted by the following target audiences:

- System Administrators
- Support Personnel
- Technical Staff

OVERVIEW OF THE MAIN SECTIONS

This guide gives an overview over all information relevant for operating Celonis and is therefore divided into the following sections:

TECHNICAL CONFIGURATION - SECURITY

This section describes the software's security features.

TECHNICAL CONFIGURATION - HIGH AVAILABILITY

This section shows which steps are necessary for Celonis to operate in a high availability environment.

TECHNICAL CONFIGURATION - LOGGING

This sections describes the logging capabilities of the application.

APPLICATION SERVER ADMINISTRATION

Since the application will be running as an operating system services, this part describes how to correctly configure it as such. It will also describe the necessary tools for administration.

PERIODICAL TASKS – ARCHIVING FILES

This section describes how to best manage the task of periodically archiving old files: the log files of the Apache Tomcat server as well as new releases of Celonis

BACKUP AND RECOVERY – BACKUP APPLICATION METADATA

Here, you will learn about how to regularly backup the metadata datastore as well as restore it in case of a failure.

BACKUP AND RECOVERY – BACKUP ANALYTICS DATABASE

Here, you will learn about how to regularly backup the analytics database as well as restore it in case of a failure.

MONITORING THE APPLICATION SERVER

This section describes how to best monitor the application server in terms of resource usage to ensure that the application can run with maximum performance and efficiency at all times.

LOGGING AND TRACING

Learn about the log files that will be created during the usage of Celonis as well as about how to manage them.

SOFTWARE CHANGE MANAGEMENT

This section describes how changes to the software are managed.

SOFTWARE UPDATE PROCEDURE

Here you can find a step-by-step guide about how to update Celonis once a new version is released.

SUPPORT DESK MANAGEMENT

This part lists the contact details of the service desk as well as the best procedure for getting in contact with it in case of problems.

TROUBLESHOOTING

Refer to this section to find a list of common issues and first instruction for soling them.

TECHNICAL CONFIGURATION - SECURITY

Celonis application provides built-in security for database connectivity. All user passwords in the application database are encoded (using SHA-256).

By default, the H2 metadata datastore is secured with a password, that is automatically generated. This password is not visible to the user and it cannot be read in any way. If you want to override this setting, you can do so by editing the App DB Settings from the "config-custom.properties" file in your installation directory.

The Celonis web access security relies on the Spring Security Framework hardening. It is recommended for you to enable and use a secure connection via HTTPS to the application right from the beginning, after the installation. This feature can be enabled as well from the "config-custom.properties" file. Upon enabling the SSL feature, you must go through the following steps:

- Set the "server.ssl" option to "true".
- If there is no keystore available, create a Java keystore. To generate a key in a local keystore, please use the Java keytool or import an existing key. A sample command for generating a new key is: "keytool -genkey -alias celonis4 -keyalg RSA -keystore keystore.jks -keysize 2048". Note that for paths on windows, you should use forward slashes (e.g. E:/celonis/my_keystore.jks). More information in the Oracle manual (http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html).
- Generate a new CSR and/or import the CRT (existing or obtained from the CA after signing the CSR) into the keystore. For more information, the same documentation from the previous step can be used.
- Provide the keystore file path using the "server.ssl.keystore" parameter.
- Specify the keystore alias using the "server.ssl.keyalias" parameter. The key alias name was provided upon the keystore creation.
- Specify the keystore password using the "server.ssl.keystorepw". This password is required to open the keystore.
- Specify the private key password using the "server.ssl.keypw". This password is required to read the private key.

During the installation process, the password for the default user "sysadmin" is requested. Please make sure that you are going to use a secure password. If there is no password specified, the installer will choose the default "$admin!" password. We do not recommend keeping the initial password for the "sysadmin" in a productive environment, thus this password should to be changed as soon as possible via the web frontend. The default password policies also force you to change the password directly after the first login. The password policies are also highly customizable from the "password-rules.properties" file. There you can enable or disable the rules and set password minimum require-

ments such as minimum length, complexity and change rate. With the number "0" the options can be set to "unlimited", for example "password.rules.last_passwords_forbidden=0" means that any old password may be reused.

Authorization in Celonis is done via the Authorization Objects. They can be used to automatically filter the dataset for users. This can be particularized for each user and dataset. With this functionality the administrator can opt for obscuring unauthorized data to be displayed to unauthorized users in such a way, that the users will not notice that they are having access to incomplete data. This grants the perfect layer of data protection and privacy for customer's data. The authorized SQL queries must be defined in the "query-definitions.xml" file.

For a secure network setup, we recommend using a dedicated server and close all ports but the ones required by our application. In the case in which another Web-Server will run in front of the Celonis Server, the server port can be bound, for example, to the localhost. This can be achieved from the "config-custom.properties" file using the "server.interface" and "server.port" parameters. Even more, all connections with the database can be encrypted. This can be done using the JDBC String, by adding the "encrypt=true" parameter.

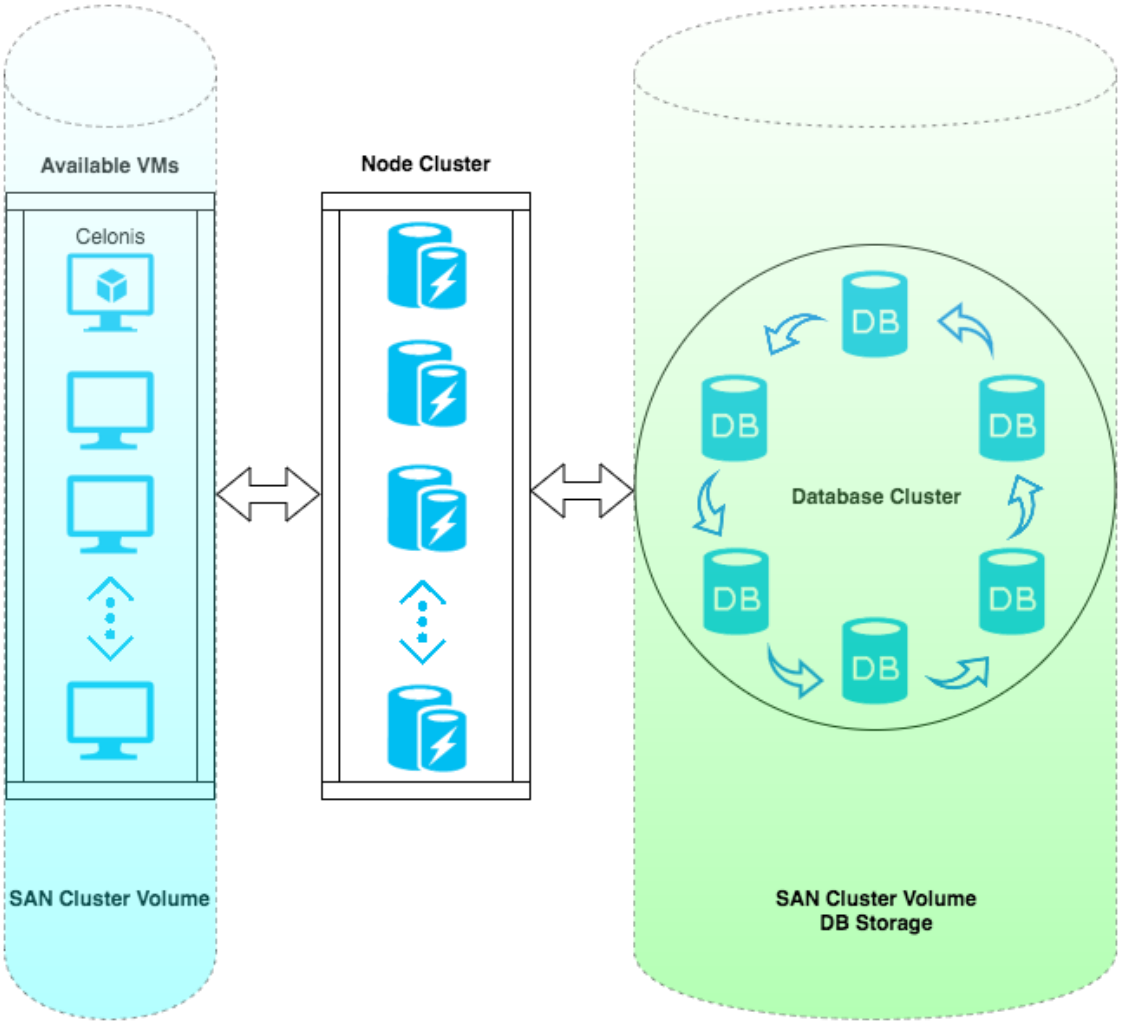
TECHNICAL CONFIGURATION - HIGH AVAILABILITY

Celonis application can be installed in a High Availability Cluster configuration in order to benefit from:

- High Application Server uptime.
- Resource scalability.
- Migration easiness.

It is recommended to use a dedicated VM Server for Celonis and to perform regular Snapshots to this VM on a remote location.

Due to the large number of infrastructure concepts only a sketch is displayed in the "HA-1" diagram below. This is not to be considered as an infrastructure design, but it should give you an overview of the key components you have to consider while using Celonis in a HA design. Networking elements and connectivity are also completely excluded from this diagram. For more information, the specific solution's and / or vendor's HA design must be consulted.


Available VMs
Celonis
Node Cluster
DB
DB
DB
Database Cluster
DB
DB
DB
SAN Cluster Volume
SAN Cluster Volume
DB Storage

HA-1

TECHNICAL CONFIGURATION - LOGGING

LOGGING FOR CELONIS

The default log level information for Celonis is "info". This only logs basic information. If you need more advanced log messages you should change this to "debug". Less information is available with the levels "warn" or "error". We do recommend keeping this at the "info" level in production environments.

Celonis also has the ability to write audit logs. The audit logger allows you to create a configuration for logging audit-relevant events. You can enable specific events by setting the audit logger configuration to true. By default, no audit log is written. To enable the configuration, copy the "audit-logging.properties.sample" file in the component_configurations folder in your installation path to "audit-logging.properties" and enable the events you are interested in. You can enable specific events by setting the options from "false" to "true". Individual options can be enabled or disabled for each of the following cases:

- Login events
- Failed logins
- User creation
- User deletion
- Group assignments
- Group creation
- Permission changes
- Object creation
- Object deletion
- Permission denied

Logging at what particular time a user has logged into the Celonis software is also possible. By default, this feature is turned off, but it can be enabled by copying the "login-logging.properties.sample" file to "login-logging.properties" and fill out the required information:

- Login_logging.enabled – either false or true
- Login_logging.database.url – as the information is saved within a database, the JDBC connection url must be entered here
- Login_loggin.database.driver – the JDBC driver used to connect
- Login_loggin.database.user – the database user with proper access rights
- Login_loggin.database.password – the database user's password
- Login_loggin.database.success_query – the query that will be executed in case of a successful login.

APPLICATION SERVER ADMINISTRATION

REQUIRED TOOLS

To successfully administrate Celonis on your Application Server, you only need a text editor.

Furthermore, the standard Linux command line tools (like tail, grep and others) will help you in accessing log and configuration files.

As Windows lacks most of those command line tools and the built in text editor is lacking features like syntax highlighting or support for UNIX-style line breaks, it is recommended to install specific tools for Windows (e.g. Notepad++, baretail, baregrep).

For administrative tasks inside the software itself a web browser is required. As the application can normally be accessed from outside the server, there is no direct need to have a web browser on the application server itself. It could however be beneficial to test connection issues, etc.

CELONIS CONFIGURATION

The Celonis server configuration takes place during the installation process. The central configuration file of Celonis is *config.properties*. This file can be found inside the root directory of the installed software, however we do not recommend manual editing. The file gets overwritten in the update process. All user custom configuration should be made in the *config-custom.properties* file.

CELONIS AS OPERATING SYSTEM SERVICE

Using the Jetty Embedded Application Server and the Apache Commons Daemon Service Runner, the Celonis application is installed as a service inside the Windows Operating System, offering the possibility to be administrated as any other regular OS service.

The Celonis service name can be customized in any particular way that it's required. The usual service name that is used by Celonis during the installation process is "Celonis". For Windows operating systems the Celonis Application Service can be configured using the following "Startup types": "Automatic (Delayed Start)" (Recommended), "Automatic", "Manual" or "Disabled".

For Linux/OSX operating systems the Celonis Application Service can be manually controlled using the "start.sh" and "stop.sh" bash scripts. Using OS specific methods, these scripts can be set to automatically run in special conditions (e.g. Automatically start the software on computer boot).

The Celonis Application Service can receive the following service commands:

- On Windows: "Start", "Stop" or "Restart".
- On Linux & OSX: controlled via the provided bash scripts.

A service restart, if needed, could also be performed by first stopping and then starting up the service.

In order to offer flexibility, Celonis does not require the operating systems service installation in order to run. The application server can also be run manually, only when it's needed, however this is not recommended in productive environments. We highly recommend using Celonis installed as an operating system service in order to benefit from easiness in administration. Operating systems services are also offering the possibility that when no longer needed, they can be uninstalled.

PERIODICAL TASKS - ARCHIVING FILES

Using Celonis for a long period of time will put you in the situation of dealing with old files. Old files will take unnecessary disk space and keeping old files mixed with current files will make the administration process more and more difficult overtime. We recommend archiving these old files and / or even set up an "Old Files Strategy" policy.

The archiving policy should consider the following cases:

CELONIS LOG FILES

The log files generated by the Celonis Server are located in the "logs" folder that resides in the root of the Celonis Server install path. The "logs" folder will contain the following log files types:

- cbpd_svc-stderr.<date>.log (Windows)
- commons-daemon.<date>.log (Windows)
- cbpd_svc-stdout.<date>.log (Windows)
- stderr (Linux)
- stdout (Linux)

As you can observe, all log files contain a date format that basically is the year, month and day of the log files creation. A new log file is generated each time you restart the software. Inside a log file, for example "*cbpd_svc-stderr.2016-01-15*", you will find only the events that occurred between server start and server stop (restart) commands. If you are going to restart the Celonis on a daily basis (highly unlikely) you will basically get each log type being generated once per day. After a year had passed, - on a Windows installation - you will have 365 "cbpd_svc-stderr.*<date>.log*" files and another 730 files summing the other types. If you want to check for errors, you should not search through log files from two-three months ago. Of course there are text filtering techniques and log files search patterns that can be used and applied (and should be nevertheless), but still going through all the log files can take a long time.

From the disk space consumption perspective, using Celonis in a large (enterprise) productive environment may generate log files up to GBs values and as files this size matter, keeping old log files will then take unnecessary disk space.

CELONIS RELEASES

Celonis gets periodic new builds that may deal with customer requested new features, tuning, customization, new web browser compatibility, bugfixes or up to date security standards. As such, we always recommend upgrading Celonis to the latest version. As the upgrading procedure describes,

old Celonis production releases (basically the install kits) should not be deleted right away, but kept as backup versions in case the customer experiences problems with the newest release (the use of older web browsers for example). At some point in time, these old versions of Celonis will take unnecessary disk space.

As we do not encourage you to delete anything unless you need to (you may not know when you will need something from the old files), we will make the following recommendations for an archiving strategy:

- Archive (.zip, .tar.gz, etc.) old log files once per month and thus keeping in the "logs" folder only log files newer than 30 days.
- Move all Celonis old software installer releases inside an "Old" folder and thus keeping only the last two releases in your current Celonis installation path (outside the "Old" folder).
- IMPORTANT NOTE: Please consult the upgrading procedure in order to be aware of all the files that are modified during upgrading to a new release – they should all be part of the old Celonis version archiving procedure. Usually we are taking care of this automatically, but there may be special releases at some point in time that will require some extra steps.
- Move all old archives to a remote location in order to free up unnecessary used disk space on the current server.

All Celonis recommendations should be treated as such and you should always consider first the digital files management policies already established by your company, if they are available.

BACKUP AND RECOVERY – BACKUP APPLICATION METADATA

In order to offer the best possible Process Mining experience Celonis is storing metadata in a data store powered by H2. For this data store there is an out of the box predefined backup policy inside Celonis.

The datastore is then automatically backed-up each night to the "*appfiles/backup*" folder in the root of the Celonis Server install path. We highly recommended to keep a remote backup of this folder. This will allow the possibility to restore the application metadata in case a disaster will occur.

The backup is set to be performed online so you do not have to worry about any Celonis downtime during this procedure.

The backup is running automated every night at exactly 3am (while the application is running) and additionally whenever the Celonis service is started. All backups taken are full backups for all application metadata.

RECOVERING FROM A BACKUP

The backup files follow the naming convention "db-backup-<yyMMddHHmmss>.h2.db" with the timestamp indicating when the backup was started. Technically, this file is a zipped version of the full application database. To restore the backup, please do the following steps:

- Identify the backup you want to restore. Use the timestamp of the backup to identify the backup you want to restore.
- Rename the backup file – append the file ending .zip to make Windows recognize the file as a zip file. After this, the file name should be db-backup-<yyMMddHHmmss>.h2.db.zip.
- Shut down the application server. Go to the services.msc, identify the service (Default: Celonis) and stop it.
- Identify the path where the active data store is located. Open the config-custom.properties main configuration file of your installation. Identify the property "database.url=jdbc:h2:DATABASE_PATH"
- At the location, there should be a file with the same name as the file contained in the zip archive.
- Create a backup of the file at the original location. To achieve this, copy the original file to the backup folder and rename it accordingly.
- Copy the file from the zip archive to the original location. Overwrite the original file with the one extracted from the backup.

- Start the application server service using the services.msc console.
- Wait for the application server to be started completely.

As a result, your backup is restored.

BACKUP AND RECOVERY - BACKUP ANALYTICS DATABASE

The Celonis Analytics Database should be backed-up on a regular basis, preferably to a remote location. You can use for your reference the Backup Policy already established by your IT Department or if such a thing is not available, you can set one up that will best suit your needs. This will allow the possibility of an Analytics Database recovery in case a disaster will occur.

When establishing a Celonis Analytics Database backup policy you must take into consideration at least the following topics:

- Database size.
- Backup destination and available backup storage space.
- The connectivity and thus the speed and throughput available from the Analytics Database Server to the Backup destination device or medium.
- Database's high usage time frames.
- Regular Schedulers or Cron Jobs that were set-up together with our Data Scientist technical personnel during the Data Integration part of the Celonis installation.

MONITORING THE APPLICATION SERVER

It is a best practice to constantly monitor the Application Server. Besides the initial minimum system requirements that are provided during the Celonis installation, additional resources must be always available, especially if the Celonis Application Server is sharing resources with other 3rd party software.

Unless specified otherwise, an operating system gets periodic updates that will increase its Disk Storage space necessity overtime. Additional Disk Storage space is also required so that the operating system can create periodic restore points. Extra Disk Storage space is to be considered if additional software, modules, libraries or features will also be installed on the Celonis Application Server machine in the near future. These are just a few cases that will make you pay attention to Disk Storage space as other factors can influence this as well.

RAM and CPU resources are also very important. Insufficient RAM and / or CPU power may lead to very poor server performance, hang-ups or can even freeze entirely the Application Server. Most applications are making use of these two resources in a dynamic way (only when necessary), so it is very important to scale them properly.

Network throughput must be taken into account if the Celonis Application Server is shared with other 3rd party software that require highly intensive and regular networking data transmission.

Taking care of everything at start is quite easy, but this is not enough in a productive environment, especially in large IT infrastructures. In such cases (but not only) you should consider using (centralized) server monitoring techniques. There are a lot of tools and features that can provide you with real-time updates regarding all server resources, depending on the server's operating system and IT infrastructure. Having access to this kind of information in real-time will help you avoid unnecessary problems related to server overburdening.

INCLUDED MONITORING FUNCTIONALITY

Apart from the Operating System's built-in monitoring capabilities, Celonis supports Java Management Extensions standard monitoring. To enable JMX, you have to configure Celonis accordingly by adding the following properties on startup. Please note that the application needs to be restarted in order to activate those changes.

-Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=<port>
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.authenticate=false

LOGGING AND TRACING

You can always make use of the Celonis Server logging system. The log files generated by the Celonis Server are located in the "logs" folder that resides in the root of the Celonis Server install path. These logs can offer you information related to:

- Starting and Stopping the Celonis Application Server.
- Exceptions of Celonis
- Queries and their timing
- Other information related to Celonis Server.

Celonis offers the possibility to configure different logging levels. The logging levels can be configured from the "config-custom.properties" file. For this guide, the most significant information is that you can assign different levels to each package: "INFO", "WARN", "ERROR", "DEBUG". Please note that changes to the logging parameters will require a restart of the Celonis to take effect.

In addition to the Celonis Server logging, Celonis logs query and application exceptions separately and makes them accessible through the application's web interface (System Settings – Exceptions). Here short-term exception information is displayed which can be very helpful when debugging applications.

NOTE: The information gathered by each of the logging systems should be used only for debugging purposes.

SOFTWARE CHANGE MANAGEMENT

New releases and support packages can be retrieved from my.celonis.de. Regardless of the type of patch you will be provided with a full installer file. The procedure for updating an installation is described in the next chapter Software Update Procedure.

When you want to promote configurations and artifacts to production, there is a built-in export/import mechanism for all transportable artifacts in the web interface of Celonis; for usage instructions, please refer to the Celonis manual. Technical configurations can be copied on a file level.

SOFTWARE UPDATE PROCEDURE

The Celonis software is shipped as an installer. The installer type depends on the Operating System it's going to be installed on.

The general update procedure is described below, however there may be several other instructions specific to a certain release. If any specific instructions should apply, they will be shipped out together with the particular release.

There will be a short downtime of the application for the duration of steps 2 to 5.

General update procedure:

1. Download the new release from my.celonis.de
2. Stop the Celonis Application Server and close all opened Windows if you are using a Windows OS.
3. Run the installer.
4. The installer will automatically bring Celonis to the latest version.
5. Start the Celonis Application Server.
6. You have successfully finished the update!

Information: If you need to check the version of Celonis while the software is not running, you can do so by viewing the *config.properties* file in the root directory of the Celonis application.

SUPPORT

Hotline: +49/89/416159677

E-Mail: servicedesk@celonis.de

Support Service: 8:30 – 17:00 (Monday – Friday)

Otherwise, please contact your personal customer advisor.

TROUBLESHOOTING

- Application not accessible:
  - Are you connected to the corporate network?
  - Do you use a proper (up-to-date) Web Browser?
  - Is the URL you are trying to access correct?
  - Is the (Database) Server running?
  - Is the Celonis Application Server running? (For your reference you can check the "Celonis as Operating System Service" chapter of this guide)
  - Login failed:
    - Is the Password correct?
    - Is the User name correct?
    - Does the User Account exist?

- Analysis is empty, no data is showing
  - Are the permission rights set correctly?
    - For the Analysis?
    - For the Data Model?
  - Was the Analysis saved after modification?
  - Were all selections reset?
  - Are the permanent filters deactivated?
  - Is the database connection successful?
  - Have you checked for any errors in data integration?

- Document/Data Model disappeared?
  - Were the permissions withdrawn?
  - Was any restore performed with an older backup? (For your reference you can check the "Backup and recovery - Backup Analytics Database " chapter of this guide)