

Have you made a backup  
yet today...?



Have you made a backup yet today...?

---

## Change Management in Engineering

**Backing up is the term generally used in the computer world for making a copy of your data in case your working data is lost. If this happens, the backup copy can be used as the basis of restoring the data to its pre-loss state. In the field of automated production, this means that downtime can be minimised after, for example, the replacement of a faulty hard disk or controller.**

With that in mind, a good data backup strategy for automated production can be defined as follows: A complete copy of the current data for each automated production device must always be kept so that the device data can at any time be restored to its most recent state. The problem is that this is not as easy as it sounds. The specific challenges here are in the following details:

- Any time: Hardware (controller, PC, hard disk) failure usually occurs without warning.
- Current data: Whether or not any information is irretrievably lost, and how much, depends on exactly how recent the backed up data is.
- Each device: It is next to impossible to predict where the next data restoration will be needed. This means that the restore data of all the components for each production-related process must be kept completely up to date at all times.

What do these challenges mean as far as data management is concerned? And how can the process of backing up be automated in order to ensure that all the data required for a full restore are always available? To answer this question properly, we first need to look closely at the often-heard question about having made a backup from two different points of view. Not only do we have to think about exactly what is meant by the question, we should also consider whether it is a question that can be answered with a simple 'yes' or 'no'.

Only then can we be sure that we will be able to meet the challenges of devising a watertight backup strategy.

### **“Have you made a backup yet today...?” What exactly is meant by this question?**

In the automated production environment, it is a question that is more than justifiable. It cannot harm to be continually reminded of the need to avoid losing data, a situation that could either directly or indirectly lead to a prolonged production stop. Having a copy of all current data can save a lot of time and money if a fault occurs. But there is a lot more to the question than it first seems. Specifically:

- Are the current state of development and parameters known for each controller and PLC on the factory floor? A PLC can only be restored to its pre-breakdown state if it is known which version of its program was being used.
- Is there a readily available copy of each device's data? It is not enough to quickly copy the current version onto whatever storage medium happens to be to hand. There needs to be central data storage so that current versions are always accessible for any member of staff.
- Has anything changed? The backup has to be updated every time something is changed. Even the smallest optimisation to the working software renders the existing copy out of date, meaning that a new one needs to be created immediately.
- Are all backup copies documented and stored within a structured and understandable system? A continuous backup regime in production generates a massive amount and diversity of data. The success and speed of the restore process directly depends on the effectiveness of the management of this data, making systematic documentation absolutely essential.
- Is the version history clear and traceable? Organised and disciplined versioning and documentation of the projects and parameters of each device enables seamless traceability of who changed what, when and why.

Considering all of the above, the pertinent question is not so much whether a backup has been made today. It is: 'Are our centrally stored project versions being continually kept right up to date?'

Have you made a backup yet today...?

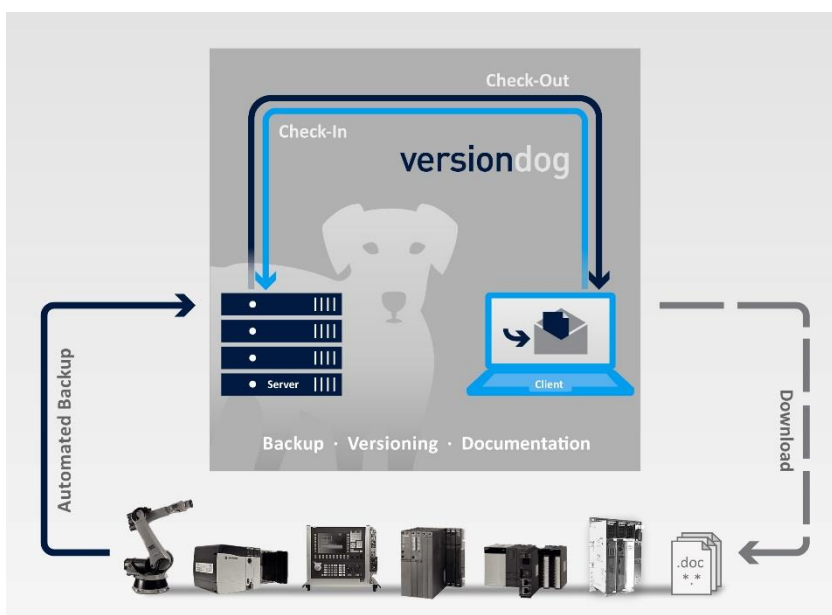
**“Yes, I have made a backup today...!” But is it really that simple? To what extent will the backup be useful if it is needed?**

Even though the question about a backup sounded so simple, it is important to think carefully before answering with a simple ‘yes’. Do you really have a backup copy of everything needed to restore each controller/device and their projects to their exact state prior to data loss? Will you be able to not only resume work straight away but also pick up precisely where you left off with ongoing maintenance, changes and optimisation?

## Example 1: Simatic S7

To make a backup of the data of an S7 controller, the usual course of action is to upload its data. Here the data contained in the controller’s memory is copied to a backup that can be used in the event of data loss. The problem is that while the data uploaded from an S7 can be downloaded back to it to restore its operation, it is in a low level language that does not include information such as comments, symbol lists and network headers. Even if it is decompiled it is not easy to follow, which makes it either impossible or extremely difficult to use it for further maintenance and development. This means that if you rely solely on a PLC upload for disaster recovery then that recovery will only be partial. The S7 can be made to continue doing what it was doing when the fault occurred, but without the documentation on the structure and processes of the program, the project (and a lot of the work that went into it) is effectively lost. In fact, to restore the project to its pre-fault state we may well have to start again from the beginning.

To summarise: Whereas it is good to have a PLC upload available to enable quick resumption of production, it only deals with the immediate problem. It is by no means a comprehensive backup strategy. For that we need to have copies of all the original versions of the project files. We also need to be 100% sure that we know which versions of the project files correspond to the program that was running on the controller when the fault occurred. Only if we have all that can we pick up the project again and run with it as if nothing has happened.



Bridging the gap between  
Information Technology (IT)  
and Operational Technology (OT) –  
a hot topic in manufacturing

## Have you made a backup yet today...?

---

### Example 2: CoDeSys-based controller

For a CoDeSys-based controller, what can we copy to use as a backup? We could copy the boot program. But what if the boot project has not been updated and does not correspond to the current online state? And even if it does, what exactly is a boot project? The answer is that it is compiled code that can be interpreted by the controller but is no longer readable by the developer. So it cannot be used for maintenance and further development. Once again, it is not enough to simply take an upload from a controller and consider the job of backing up done. To ensure that an up-to-date and usable backup for a CoDeSys controller is always available, the version of the compiled boot program on the controller must be compared at regular intervals with the online version of the boot program, and this in turn must be compared with the centrally stored project files. If you can be certain that all three correspond to one another, then your production is safeguarded as follows:

- The current state of the controller software is known
- If the controller needs to be rebooted, you can be sure that the same program as the one that was running will be reloaded
- The current project files are available centrally at all times and could be compiled at any time to produce an up-to-the-minute restoration copy of the controller software or used for the purposes of ongoing maintenance and optimisation

To answer the question about whether or not a backup has been made today with an unequivocal 'yes', then all three of the above points need to apply. A copy of the controller software in itself does not even cover one point. Without a dependable system of documentation, you can never be sure that it is a copy of the most recent software. Without a copy of the project files you cannot continue to work on the project. Clearly we need a comprehensive and all-encompassing strategy for each controller and each piece of hardware.

It is also clear that the system will have to involve much more than simply making PLC uploads.

### Requirements for a watertight backup system

If we bear in mind the considerations so far in this article and add a number of necessary practical considerations, we can derive the following set of requirements for daily backups in automated production:

- The system must always contain the most recent versions and they must be readily accessible at any time.
- Restore data has to be handled in the same way for all makes and models of devices.
- The system has to check at regular intervals whether the software active on the controllers corresponds to the most recent version of the project files. Only then can we be certain that the current backup copy is valid.
- If a difference is detected between the current project files and those that were used to compile the currently active controller software, a report needs to be automatically generated prompting the user to react accordingly.
- The system must be able to show the detected differences in detail.
- The system must have backup strategies for common devices used in automated production (PLCs, HMIs, robots, custom PC-based systems, intelligent field devices) and be able to handle common protocols (UNC, FTP, SSH/SSL) so that backups can be made of working directories, hard disks and the data stored on any type of computer system.

Have you made a backup yet today...?

---

## A software-based solution

If we consider the points mentioned in this article and perhaps redefine our ideas about what a 'backup' is – or should be – then it quickly becomes clear that we need to be very careful when we decide on a solution. A software-based solution, possibly an automatic one, would seem to be the obvious choice. But it will need to function in practice with the enormous number of variables that exist in the everyday working world of automated production. That is a big ask. But it is the area in which the company AUVESY specialises. Their versiondog software is a solution that has been developed with a clear understanding of exactly what needs to be backed up, exactly when it needs to be backed up and exactly how it can be backed up – and used if necessary – in real world conditions.

**[www.versiondog.com](http://www.versiondog.com)**

### Author:

Dr.-Ing. Tim Weckerle, AUVESY,  
Head of Development and Product Management

### Authorized Distributor:

Industrial Software Solutions

19909 - 120th Ave NE, Suite 106  
Bothell, WA 98011 | USA  
Tel: +1 855.724.8570

Email [sales@issconnectivity.com](mailto:sales@issconnectivity.com)  
Web [www.issconnectivity.com](http://www.issconnectivity.com)

