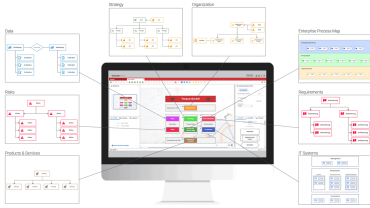


Introduction to Scheer PAS Business Modeler



Scheer PAS Business Modeler is an easy to use, innovative and cloud-based modeling tool that provides the ability to design business processes via a graphical, browser-based user interface. It is used in many projects to support the journey to complete process digitization.

The intuitive web-based user interface enables you to quickly display your own processes, process maps, organizational and IT data structures, including requirements and risks. This information can be used as a "single point of truth" for a variety of projects, from classic process management to ISO certification activities to the support of IT system implementations or process automation initiatives. The functional scope of the Business Modeler, which was developed especially for medium-sized companies, results from many years of experience of the Scheer GmbH consultants in business process management projects.

The start to process modeling is easy due to the cloud-based architecture and the intuitive handling in the browser. All necessary methods like BPMN 2.0, EPC as well as data and organizational views are supported. The modeling tool, which can be individually adapted to different use cases, helps to improve internal processes as well as communication between departments and IT. The **Scheer PAS Business Modeler** forms the basis for subsequent process automation in the **Scheer Process Automation Suite (PAS)**.

The new component can not only be used to model your own facts, but also to directly use the contents of **Scheer performanceREADY** in projects. **Scheer performanceREADY** is the modular system of Scheer GmbH for SAP solutions and efficient projects.

Scheer PAS Business Modeler benefits:

- fast access
- intuitive
- customizable
- executable (base for process automation in **Scheer PAS**)
- flexible (allows agile work)
- timesaving (allows collaborative work and automated evaluations)
- cost effective (low cost through cloud, jumpstart for straightforward projects)