Industrial plant software is implemented in the programming languages of IEC 61131-3. As plant software needs to perform many tasks, it is often highly complex and typically characterized by a monolithic structure. Whereas for high-level programming languages, commonly occurring problems are solved using design patterns, such general reusable solution alternatives are not yet available for IEC 61131-3. Thus, an approach for statically analysing the plant software and visualizing the software units' complexity and interconnectedness is proposed in this paper. Furthermore, basic software design patterns are introduced and, subsequently, their appearance within plant software is evaluated using industrial code and interviews with experts. By that, a rst step towards providing design patterns for IEC 61131-3 is made.