Pfizer Global Research and Development, Sandwich. Technology Development

LECIS - based Laboratory Automation in the Pharmaceutical Industry

Dorothe Steidinger , Dr. Sam Sydenham, Dr. Alistair Swanson

Introduction

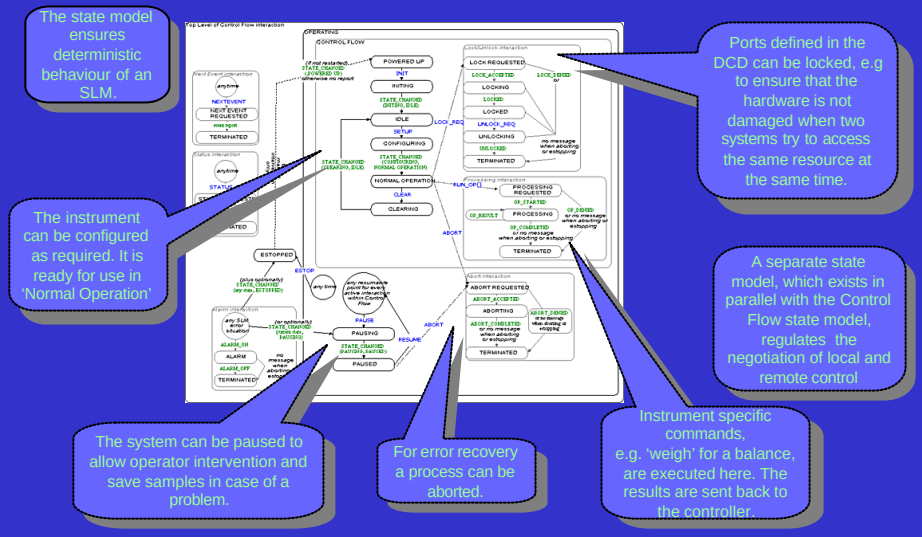
A common problem for integration of laboratory systems is the prevalent use of proprietary software. In many cases this does not make it easy to interface off-the-shelf hardware from different manufacturers. For a fully automated laboratory, however, not only the integration of individual hardware components into a standalone system is required, but also the integration of multiple 'standalone' systems and a LIMS system for data management.

LECIS

a definition of a common remote control interface for laboratory equipment which describes the dialogue between the control unit (TSC) and the instrument (SLM) that is necessary to co-ordinate operations. Some commands are set by the LECIS (e.g. 'pause', 'clear', 'resume'), others are instrument specific. The latter are defined in the device capability dataset (DCD), which is supplied by the vendor. The LECIS has been published as ASTM E1989-98 and an adaptation is being developed by the OMG making use of the CORBA.

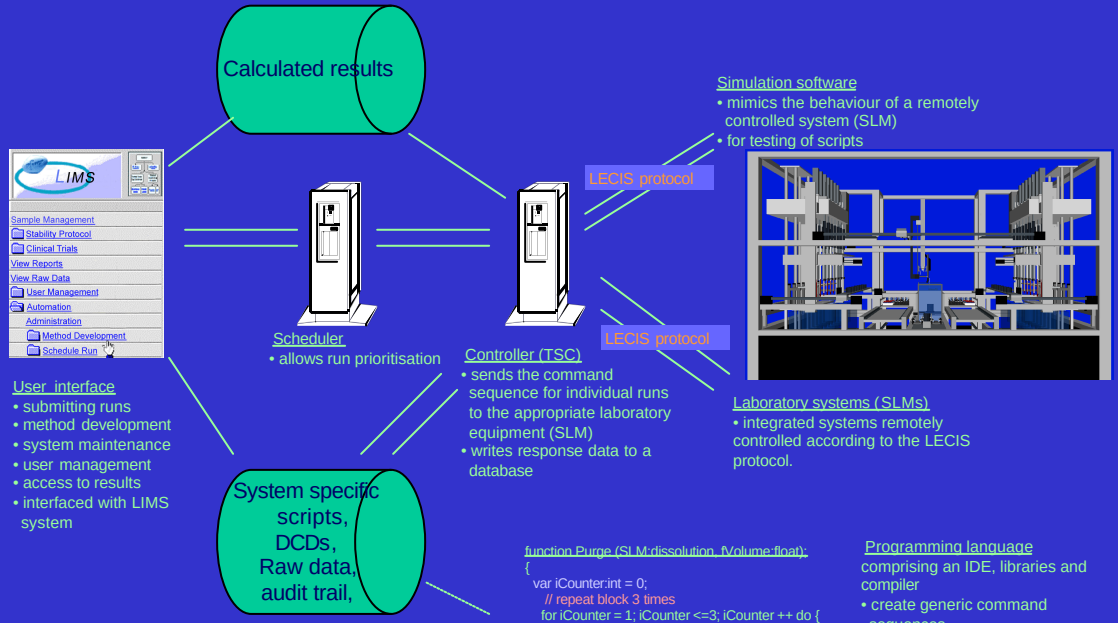
Benefits:

- "plug-and-play" of equipment with a common interface that is independent of the vendor
- computing platform independent
- programming language independent
- simplifies instrument integration, thus reducing costs



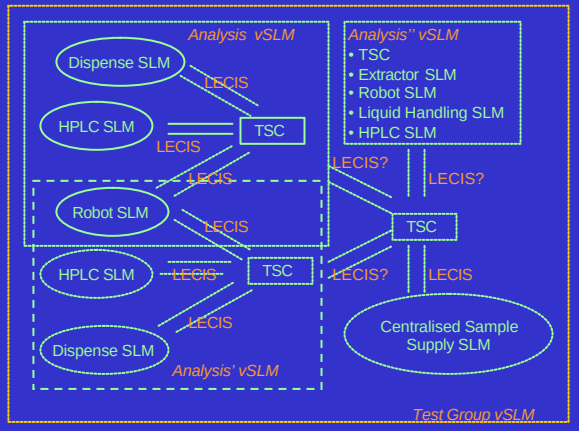
Project

At Pharmaceutical Sciences, Sandwich, a project is underway to automate a set of analyses frequently performed during the drug development process. Hardware modules with a common sample supply interface, as well as the use of the LECIS protocol for the control of these modules, will ensure that the modules can be integrated for ease of use and be upgraded to keep up with the changing needs of the department.
The aim is to develop a complete package that not only meets regulatory requirements, but also contains tools to aid the programmer, minimises the training overhead for users through a single access point, and increases throughput through centralised sample management.
The figure below shows a schematic overview of the project.




```
function Purge (SLM:dissolution_fVolume:float);
{
  var iCounter:int = 0;
     // repeat block 3 times
    for iCounter = 1; iCounter <=3; iCounter ++ do {
      SLM.Aspirate (fVolume);  //aspirate 5ml liquid
       }
}
```

Programming language
comprising an IDE, libraries and compiler
- create generic command sequences
- scheduling decisions at run time

Challenges

A number of challenges have to be overcome to achieve the goal of an automated laboratory:
- the development of a system architecture that can be scaled up
- complex systems that cope with shared hardware
- Limitations of the standard itself, e.g.
  - * the lack of synchronous alarms make targeted error handling difficult
  - * the DCD is not defined by the standard and does not fully meet its needs
  - * the standard is perceived as over-complicated by many vendors (e.g. through the programming overhead created by the Nextevent and Lock/Unlock interactions), therefore not many products are available.