Funding Scheme: THEME [ICT-2007.8.0] [FET Open]

Paving the Way for Future Emerging DNA-based Technologies: Computer-Aided Design and Manufacturing of DNA libraries

Grant Agreement number: 265505

Project acronym: CADMAD

Deliverable number: D3.4

Deliverable name: Report on the development of low level drivers for Hamilton Star and Tecan platforms

Contractual Date[1] of Delivery to the CEC: M24
Actual Date of Delivery to the CEC: M24
Author(s)[2]: Ellis Whitehead, Joerg Stelling
Participant(s)[3]: ETHZ
Work Package: WP3
Security[4]: Pub
Nature[5]: R
Version[6]: 1.0
Total number of pages:

1 As specified in Annex I
2 i.e. name of the person(s) responsible for the preparation of the document
3 Short name of partner(s) responsible for the deliverable
4 The Technical Annex of the project provides a list of deliverables to be submitted, with the following classification level:
Pub - Public document; No restrictions on access; may be given freely to any interested party or published openly on the web, provided the author and source are mentioned and the content is not altered.
Rest - Restricted circulation list (including Commission Project Officer). This circulation list will be designated in agreement with the source project. May not be given to persons or bodies not listed.
Int - Internal circulation within project (and Commission Project Officer). The deliverable cannot be disclosed to any third party outside the project.
5 **R (Report):** the deliverables consists in a document reporting the results of interest.
P (Prototype): the deliverable is actually consisting in a physical prototype, whose location and functionalities are described in the submitted document (however, the actual deliverable must be available for inspection and/or audit in the indicated place)
D (Demonstrator): the deliverable is a software program, a device or a physical set-up aimed to demonstrate a concept and described in the submitted document (however, the actual deliverable must be available for inspection and/or audit in the indicated place)
O (Other): the deliverable described in the submitted document can not be classified as one of the above (e.g. specification, tools, tests, etc.)
6 Two digits separated by a dot:
The first digit is 0 for draft, 1 for project approved document, 2 or more for further revisions (e.g. in case of non acceptance by the Commission) requiring explicit approval by the project itself;
The second digit is a number indicating minor changes to the document not requiring an explicit approval by the project.

Abstract

To enable robot programming with high-level commands (e.g.,'do PCR'), it is essential to provide interfaces and drivers that enable an appropriate control of the robot at the low level (e.g., 'aspirate volume x from well y'). In task 3.3., corresponding drivers of low-level functionalities for the robot platform and for devices attached to the robot platform – such as PCR machines – were developed. Efforts focused on the Tecan platform for two reasons, namely (i) to develop a generic framework by directly coupling software development and testing on an existing robot, and (ii) to specifically improve quality control capabilities that proved to be a critical issue for overall project progression (deviations from planned work for this reason). Device drivers for all relevant functionalities for CADMAD (liquid handling, PCR, measurement devices, and associated devices) were developed and validated experimentally. Specific access to measurement devices of the robot was enabled to improve quality control. The framework proved suitable in practice, and corresponding extensions to Hamilton platforms are envisaged.

Keywords[7]**:**

Liquid handling robot, device driver, quality control

1. Introduction

a. Aim / Objectives

The general aim of WP3 is to automate DNA processing using advanced robotic platforms, among others, by developing a robot programming language (Robo-Ease) that allows for a device-independent specification of molecular biology protocols at a high level (akin to high-level, potentially object-oriented programming languages in standard computer programming). An essential building block to achieve this objective is to develop configuration capabilities (hardware and low-level drivers) that allow users to automatically and dynamically generate and execute Robo-Ease scripts according to the systems requirements. The present report deals with the development of low-level drivers in this context.

b. State of the Art

Current liquid handling robot programming offers two basic options: (i) graphical assembly of robot programs using manufacturer-supplied software that is typically configured for a single robotic platform (liquid handling system and associated devices) with the disadvantages of a lack of portability of protocols between robots of different type and / or configurations, and of severe restrictions for the implementation of complex molecular biology protocols such as those used in CADMAD. (ii) low-level programming with the same limited portability, and poor possibilities for debugging, especially when complex protocols are concerned (see examples below).

c. Innovation

The main innovative approach for robot programming consists in measures to move unavoidable complexity to the robot programming language, instead of requiring users to specify all details of complex protocols. Low-level drivers as detailed below provide the essential interface for this purpose because they enable the specification and decomposition of higher-level commands to achieve machine-executable sequences of command. This ultimately enables a consistent representation of data and programming objects, and an easy development of concrete robot control programs, for instance, by accessing databases for the specification of robot configurations, reagents, and protocols.

[7] Keywords that would serve as search label for information retrieval

2. Implementation

Conceptually, we distinguished between two types of low-level drivers: a driver to control the overall robot platform, and drivers to control specific devices attached to the robot. Both kinds of drivers were written in a Java-compatible language (preferably Scala). They are used as plugins by the overall system for controlling a robot.

The robot platform drivers are responsible for producing output that can be executed by the robot platform. This involves taking generic low-level commands and turning them into platform-specific commands. For example, the robot driver may be supplied with a low-level command to aspirate liquid from a single well such as follows:

```
AspirateToken(List(TipWellVolumePolicy(0, P1(A01), 50ul, PipettePolicy(Wet, WetContact)))),
```

where the parameters for liquid-handling operation (values for selecting tip and well, liquid volume, and pipetting protocol) are given in the list of arguments.
The Tecan platform driver developed here would translate the above generic command into something like the following (depending on the robot configuration):

```
Aspirate(1,"Wet","50",0,0,0,0,0,0,0,0,0,0,0,17,0,1,"0C0810000000000000",0,0);
```

Work on implementing drivers for the Hamilton platform was postponed, because we deemed the quality control issues to take precedence.

3. Results

The following device drivers (where 1-4 control the liquid handling robot, 5-11 refer to associated devices) have been implemented on the Tecan platform; function of the drivers was validated in experimental tests on the integrated robotic platform:

1) *Tecan platform driver*. The primary responsibility of the Tecan platform driver is to generate Evoware (Tecan-specific software for robot control) scripts for a specific robot setup. The platform driver reads configuration files from an existing Tecan installation in order to gather most of the information it needs about the workbench and available devices. Given this information, and possible additional user-supplied configuration parameters, the platform driver provides various default values for missing parameters to low-level device commands.
2) *Tecan liquid handling*. The liquid handling drivers provide functions to aspirate, dispense, mix, wash, get and drop tips.
3) *Tecan liquid level detection (low-level)*. Evoware provides inaccurate volume measurements due to its limited ability to accommodate various well shapes. Using the sensor information directly, the driver extracts raw level detection data in order to improve the estimation of liquid volumes, in particular for quality control purposes (see Example 1 below).
4) *Tecan robotic arm*. The arm driver provides capabilities to transfer plates between racks and handle plate lids. For multi-arm robots, an arm can either be specified or otherwise the platform driver will choose the best arm to use.
5) *Tecan shaker*. The shaker driver is used to shake plates. Parameters such as displacement and frequency can either be specified or defaults can be obtained from the platform driver.
6) *Tecan filter*. The filter driver lets us perform filtration by use of a special filter plate and a suction pump.
7) *Centrifuge*. The centrifuge driver controls operation of a centrifuge. This is one of the more complex devices to control because of the requirement that plates be placed symmetrically in the centrifuge with approximately equal weights. These considerations are taken care of by higher-level commands, and the low-level driver merely handles the tasks of making specific locations on the centrifuge available to the robotic arm and of running the centrifugation with supplied parameters or defaults.
8) *Plate readers*. The plate reader drivers control our absorbance readers. They can produce control files for Biorad readers and parse Infinite 200Pro XML output files (see Example 2 below).
9) *Tecan sealer*. The sealer driver operates the sealing of a plate by adhesion of a plastic film over the top of the plate for air-tight closure of the wells.
10) *Tecan peeler*. The peeler driver operates the peeling off of a sealing film from the top of a plate.
11) *T-Robot thermal cycler*. The thermal cycler driver operates a PCR thermal cycler. An identification code for a specific PCR program must be specified as input. If there are multiple cyclers on a bench, the user can choose one

or let the platform driver choose a default. The driver can also produce the low-level program specification that can be loaded on the T-Robot by the user.

Most devices on standard robotic platforms do not provide feedback. However, two notable exceptions are the liquid level detectors and the plate readers. The following two examples illustrate how this feedback is read back in such that it can be used for on-line quality control of the robot operation.

Example 1: Liquid level detection

As touched upon above, Evoware does not produce usable liquid level measurements for most plate models employed in CADMAD. To take advantage of the robot's ability to detect liquid levels, the low-level driver reads in the raw data from the robot via Evoware's execution log. The log file contains a mixture of descriptive output and raw low-level robot codes. Our parser uses the descriptive entries to track aspirations and dispenses in each well, and parses the robot codes to get the raw "z-level" measurement values ("z-level" refers to the vertical position of the tip at the height at which it detects liquid). By combining all this data, we can find the correlations between liquid volume and z-level. The following two lines are excerpted from a log file produced during a quality control run:

```
> C5,RPZ0
- C5,0,1380,1384,1386,1393,2100,2100,2100,2100
```

Given the prior information in the file about which wells the levels were read from and the volumes that were dispensed into the given wells, the following tab-separated entries are generated:

tip	loc	row	col	step	z	vol	dz	dvol
1	10,2	5	12	2	1379	260.00	32	110.00
2	10,2	6	12	2	1385	270.00	36	120.00
3	10,2	7	12	2	1385	280.00	39	130.00
4	10,2	8	12	2	1394	300.00	45	150.00

This allows the user (or higher-level commands) to analyze aggregate statistics about tips, bench locations, well positions (row and column), steps in multi-step mixtures, z-level, well volumes, as well as changes in z-levels and well volumes. The probability distribution that correlates z-level to volume enables substantially more accurate liquid level measurements.

Example 2: Absorbance reader

The Infinite 200Pro reader produces XML output that can be used as input to another procedure for analysis or to determine future robot operations. Take for example the following XML snippet for a single well readout from a quality control run:.

```
<Well Pos="A1" Type="Single">
         <Single LabelId="1" Status="Measured">0.9672</Single>
         <Single LabelId="2" Status="Measured">0.0384</Single>
         <Single LabelId="3" Status="Measured">0.9288</Single>
</Well>
```

This information gets parsed along with information about the originating protocol, producing an entry in a tab-separated file. These entries contain enough information to allow for complex analyses over many runs over arbitrary spans of time. For the above XML entry, the parser produces the following output (presented in an alternate form for the sake of readability):

```
test id: read10times1Jb
date: 2012-10-08
script: ellis_dye_read10times.esc
site: 10,2
plate model: 96 Well Microplate
liquid class: Water free dispense
base volume: 190
base concentration: 0
volume: 10
concentration: 0.8
tip: 3
tip volume max: 0
tip volume: 10
multipipette: 0
row: 1
col: 1
readout: 0.9288
```

The driver / parser combination generates one such entry for every well in each read-out, allowing us to group our data by tip, plate model, liquid class, bench site, and more. Some of the resulting graphs are shown in the report for D3.3.

4. Conclusions

Device drivers for all relevant functionalities for CADMAD operating on Tecan robotic platforms (liquid handling, PCR, measurement devices, and associated devices) were developed and validated experimentally. Specific access to measurement devices of the robot was enabled to improve quality control. The framework proved suitable in practice, and corresponding extensions to Hamilton platforms are envisaged.

5. References

Tecan Evoware: http://www.tecan.com/platform/apps/product/index.asp?MenuID=1458&ID=1084&Menu=1&Item=21.7.1

6. Abbreviations

List all abbreviations used in the document arranged alphabetically.

DNA	Deoxyribon Nucleic Acid
PCR	Polymerase Chain Reaction
XML	Extensible Markup Language