Delta
July 5, 2018

Software Transfer Document
Version 1.0

**Project Team**
D.P. van den Berg 0949036  
R.T. van Bergen 0938857  
D.J.C. Dekker 0936100  
S. van den Eerenbeemt 0954445  
J. Mols 0851883  
B.F. Rongen 0858160  
B.W.M. van Rooijen 0895073  
R.P. Schellekens 0944330  
A.A. Vast 0854060  
G. Walravens 0904152  
S. Wessel 0941508

**Project Managers**
S.P.O. Oostveen  
A. Rajaraman

**Project Supervisor**
dr. N. Zannone

**Customer**
dr. L. Genga
Abstract
This document is the Software Transfer Document (STD) for the Delta extension. The Delta extension of the APD tool allows users to perform experiments on process models and event logs. Along with user management, Delta implemented further experiment and project management functionality. This STD describes how to build and install the Delta extension, such that the product can be successfully transferred to the customer. The document also contains a report on the acceptance test. This document complies with the ESA Software Engineering Standard [1].
## Document Status

### General

**Title of document**: Software Transfer Document  
**Document identifier**: Delta.STD/1.0  
**Version**: 1.0  
**Authors**:  
- D.P. van den Berg  
- R.T. van Bergen  
- D.J.C. Dekker  
- S. van den Eerenbeemt  
- J. Mols  
- B.F. Rongen  
- B.W.M. van Rooijen  
- R.P. Schellekens  
- A.A. Vast  
- G. Walravens  
- S. Wessel

### History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Author(s)</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2018-07-01</td>
<td>D.P. van den Berg</td>
<td>Describe building procedure.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>S. Wessel</td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>2018-07-04</td>
<td>D.P. van den Berg</td>
<td>Describe environment installation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>S. Wessel</td>
<td></td>
</tr>
</tbody>
</table>
## Document Change Records

<table>
<thead>
<tr>
<th>Section</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Describe environment installation.</td>
</tr>
<tr>
<td>5</td>
<td>Acceptance test report.</td>
</tr>
<tr>
<td>6</td>
<td>Software problem reports.</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Purpose

The Software Transfer Document will describe what steps need to be taken to build and install the Delta extension. This information is needed for the customer, but will also help any other party interested in setting up the system. Along with elaborating on the software problems that still exist and the software/hardware improvements made, this STD will also report on the acceptance test to give a view of the developed state of the system.

1.2 Scope

Team Delta is a Software Engineering Project group working for the TU/e and Laura Genga, who made the APD tool with a group of researchers from the Marche Polytechnic University. The delivered product is an extension of this APD tool, its purpose being to allow users to perform experiments on process models and event logs. Alongside the addition of user management functionality, users of the tool can manage their projects and experiments in a personalized web interface. Users can register for accounts, view and edit their account information and (with a registered account) create and manage projects. Experiments can be created inside a project, which can generate anomalous subgraphs and partially ordered anomalous patterns from the project’s process model and event log. These experiment results can be viewed side by side, and can be interacted with through a user interface.

1.3 List of definitions and abbreviations

1.3.1 List of definitions

For any definitions specific to this product, the list of definitions in Section 1.3.1 of the URD [9] can be referenced.

1.3.2 List of abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APD</td>
<td>Anomalous Process Discovery</td>
</tr>
<tr>
<td>AT-E</td>
<td>Acceptance Test Experiment Management</td>
</tr>
<tr>
<td>AT-P</td>
<td>Acceptance Test Project Management</td>
</tr>
<tr>
<td>AT-U</td>
<td>Acceptance Test User Management</td>
</tr>
<tr>
<td>ATP</td>
<td>Acceptance Test Plan</td>
</tr>
<tr>
<td>SRD</td>
<td>Software Requirements Document</td>
</tr>
<tr>
<td>STD</td>
<td>Software Transfer Document</td>
</tr>
<tr>
<td>URD</td>
<td>User Requirements Document</td>
</tr>
</tbody>
</table>

1.4 List of references


1.5 Overview

In Section 2 the build procedure for the Delta extension of the APD tool is described. Then, in Section 3, the procedure for how to install this extension is given. Section 4 contains a list of all the configuration items that are to be transferred. A short summary of the acceptance test reports is given in Section 5. Section 6 is the last section with content and contains the list of software problem reports that were raised during the transfer phase. Sections 7 and 8 list the software change requests and software modification reports respectively. Since neither of these are applicable to this project, they are left empty.
2 Build procedure

Both the APD tool and the Delta extension are built using PHP and JavaScript with a MySQL database. As PHP and JavaScript are scripting languages, no building or compiling is necessary.
3 Installation procedure

This section describes the installation procedure for a deployment host and a development envir-
onnement. We expect Ubuntu 14.04 LTS Desktop to be installed on the development environment
and Ubuntu 14.04 LTS Server to be installed on the deployment host. Whenever angled brackets
(‘<‘ and ‘>’) are used, the contents are supposed to be replaced with the appropriate value described
within the brackets. Commands executed using the *sudo* prefix are expected to be run by a user
that has super user privileges.

Optionally, the development environment might be installed on a virtual machine—VMWare Work-
station Player is recommended. In this case, the source code can be modi/uniFB01ed on the host and
mapped to the virtual machine through the use of shared folders.

3.1 General installation

The following instructions apply to both the deployment and development environment. The in-
stallation procedure consists of the following steps.

1. Update the aptitude package list and upgrade existing packages
   
   $ sudo apt-get update & & sudo apt-get upgrade

2. Install Apache2 web server
   
   $ sudo apt-get install apache2

3. Install MySQL server
   
   $ sudo apt-get install mysql-server
   
   when prompted for the MySQL root user password, set a secure password.

4. Setup MySQL server
   
   $ mysql_secure_installation
   
   Follow the interactive script to remove the guest user and test database.

5. Install PHP5
   
   $ sudo apt-get install php5

6. Install Graphviz extension

   (a) Install extension.
   
   $ sudo apt-get install libgv-php5
   $ sudo apt-get install graphviz

   (b) Enable extension.

   Create a file in /etc/php5/mods-available named gv.ini with the following content:
   
   ; configuration for php graphviz module
   ; priority=20
   extension=gv.so

   Execute the following commands to create a symlink to the ini file.
   
   $ sudo ln -s ../../ mods-available/gv.ini /etc/php5/cli/conf.
   ↔ d/20-gv.ini
   $ sudo ln -s ../../ mods-available/gv.ini /etc/php5/apache2/
   ↔ conf.d/20-gv.ini
7. Install the MySQL PHP extension
   $ sudo apt-get install php5-mysqlnd

8. Install java 1.8
   $ sudo add-apt-repository ppa:webupd8team/java
   $ sudo apt-get update
   $ sudo apt-get install openjdk-8-jdk

9. Get a copy of the source code. Let <source-root> denote the root path of the source.

10. Setup the Apache www root folder. Copy (or symlink through a shared folder if using a VM) the <source-root>/Esub folder in the /var/www folder.

    Then change the root of apache to /var/www instead of /var/www/html. This can be done by editing the DocumentRoot setting in the /etc/apache2/sites-available/000-default.conf file.

    Copy the <source-root>/database folder in the source root to some non-public folder. Let the path to the database folder be called <db-folder-path>.

    Restart Apache:
   $ sudo service apache2 restart

11. Install subdue and autodue
    (a) Create destination folder for legacy tool
        $ sudo mkdir -p /opt/subdue-master/bin

    (b) Acquire and install the subdue package included in source:
        <source-root>/devutils/subdue-5.2.2.zip.

        The package can also be obtained from the Washington Sate University Artificial Intelligence Laboratory website\footnote{http://ailab.wsu.edu/subdue/software/subdue-5.2.2.zip}.

        $ sudo apt-get install unzip
        $ sudo unzip subdue-5.2.2.zip
        $ cd subdue-5.2.2/src/
        $ sudo make
        $ sudo make install
        $ cd ../bin
        $ sudo cp * /opt/subdue-master/bin

    (c) Acquire and install the autodue package included in source:
        <source-root>/devutils/autodue-master.zip.

        The package can also be obtained from kendejibaozi's Github page\footnote{https://github.com/kendejibaozi/autodue/archive/master.zip}.

        $ sudo apt-get install libmysqlclient-dev
        $ sudo apt-get install openssl
        $ sudo apt-get install libssl-dev
        $ sudo apt-get install libxml2-dev
        $ sudo unzip autodue-master.zip
        $ cd autodue-master/autodue/src/
        $ sudo make autodue
        $ sudo cp autodue /opt/subdue-master/bin/

12. Setup the legacy APD tool MySQL user.

\footnote{http://ailab.wsu.edu/subdue/software/subdue-5.2.2.zip}
\footnote{https://github.com/kendejibaozi/autodue/archive/master.zip}
(a) Run the following command to open an SQL shell.

$ mysql -u root -p

(b) Enter your root MySQL password.

(c) Run the following sql commands:

```
CREATE USER 'esub'@'localhost' IDENTIFIED BY '18_eSub_tue';
GRANT ALL PRIVILEGES ON *.* TO 'esub'@'localhost';
FLUSH PRIVILEGES;
```

13. Setup the Delta extension MySQL user and database.

(a) Run the following command to open an SQL shell.

$ mysql -u root -p

(b) Enter your root MySQL password.

(c) Run the following sql commands:

```
CREATE USER 'delta'@'localhost' IDENTIFIED BY '<delta-db-password>';  
CREATE DATABASE delta; 
GRANT ALL PRIVILEGES ON *.* TO 'delta'@'localhost'; 
FLUSH PRIVILEGES;
```


(a) Go to `<db-folder-path>`.

(b) Make sure `dbm.sh` is executable:

$ sudo chmod +x dbm.sh

(c) Execute `dbm.sh`:

$ ./dbm.sh delta <delta-db-password>

15. Set the credentials in the Delta extension.

Go to `<source-root>/Esub/delta/database` and open `database.php`. Change the second parameter of the `DatabaseCredentials` instantiation to the username of your mysql account: `delta`. Change the third parameter to the password of the account, `<delta-db-password>`. Save and exit.

16. Setup the host name. Go to `<source-root>/Esub/delta/settings` and open `settings.php`. Change the function `getHostName()` so it returns the host name you wish to use.

17. Change `sh` from `dash` to `bash`

$ sudo rm /bin/sh
$ sudo ln -s /bin/bash /bin/sh

### 3.2 Deployment additional installation

The following procedure applies only to the deployment setup.

1. Set up the firewall by running the following commands:

$ sudo ufw allow http
$ sudo ufw allow https
$ sudo ufw allow ssh
$ sudo ufw enable
3.3 Development additional installation

The following procedure applies only to the development setup.

1. Install and enable the Xdebug package for php.
   (a) Install the package
       $ sudo apt-get install php5-xdebug
   (b) Enable the package.
       Add the following to the bottom of /etc/php5/apache2/php.ini.
       ```ini
       [xdebug]
       zend_extension="/usr/lib/php5/20100525/xdebug.so"
       xdebug.remote_enable=1
       xdebug.remote_port=9000
       xdebug.remote_connect_back=1
       ```
       Restart Apache:
       $ sudo service apache2 restart
       Debugging is now enabled on the server.

2. We recommend PhpStorm to use the Xdebug functionality. This can be enabled in
   File → Settings → Languages & Frameworks → PHP → Debug
   For a Virtual Machine setup, you can use the 'Remote Web Server' option (when running PhpStorm on the host) after installing an OpenSSH server on the client:
   $ sudo apt-get install openssh-server
4 Configuration item list

In this Section all items that Delta will be delivering will be listed. Documents and test plans are provided in a pdf format. Providing the software will be done by delivering the source code.

4.1 Documentation

1. URD [9]
2. SRD [5]
3. ADD [4]
4. SUM [7]
5. STD [6]

4.2 Test plans

1. UTP [8]
2. ATP [3]

4.3 Software

1. APD tool with the Delta extension
The first acceptance test was performed on June 22nd 2018, from 14:00 until 16:00. The procedure for this acceptance test is described in version 0.40 of the ATP [2]. During this test certain problems were discovered with the Delta extension of the APD tool. These problems caused the following test cases to fail: AT-U18, AT-E22, and AT-E23. The causes for these problems were quickly resolved after the first test.

The second and final acceptance test took place on July 3rd at 14:00 and the procedure for this acceptance test is described in the ATP version 1.00 [3]. During this acceptance test only the tests failed during the first acceptance had to be executed, as agreed with the product owner. Test cases AT-U18, AT-E22 and AT-E23 were executed successfully, completing the acceptance test for the Delta extension.
6 Software problem reports

6.1 Experiment deletion

The Delta extension of the APD tool supports the creation of projects and experiments within these projects. These projects can be deleted and upon deletion the experiments are removed too. The deletion of single experiments has not been implemented due to time constraints. Having this functionality would give the user more power over their projects and prevent the amount of experiments in a project getting too large. The backend for experiment deletion is already present within project deletion, this could be extended and connected with the frontend to allow experiment deletion.

6.2 Anomalous subgraph discovery

For any experiment a user can run the anomalous subgraph discovery phase and view the results once the phase is completed. On the first run of anomalous subgraph discovery phase this works fine, but problems occur when the phase was already completed once and is started again. The second run on the same experiment, will result in a lower number of subgraphs being loaded and different results from the first run. Whether this problem lies with the Delta extension or was already present in the APD tool is not clear.

6.3 Experiment status live updating

The project page shows the status of all experiments and the status of a phase within a specific experiment. After starting a phase the status is updated to running, but after that the status is never automatically updated and will only change on page refresh. Due to time constraints this live updating was not implemented, but this feature could improve the feedback for the user on the status of experiments.

6.4 Results in new tab

On the results page a user can select a set of subgraphs and open these in a new tab. Usually this is displayed correctly, but there are certain cases where an error occurs in the drawing of the subgraphs. For example, subgraphs that are not properly connected or an extra place being drawn that was not a part of the selected subgraphs. Due to time constraints and the rarity of the bug the cause could not be found.

6.5 SQL injection event log

It is currently possible to inject SQL code by setting the name of the traces in the event log file you upload while creating a project. The Delta extension is protected from this but on the APD tool this injection of SQL code is possible. Because the Delta extension passes the event log on to the APD tool in order to use it's functionality, this injection is still possible.
7 Software change requests

This section is left empty on purpose as no software change requests were raised during the transfer phase.
8 Software modification reports

This section is left empty on purpose as no software modification reports were completed during the transfer phase.