

|  |
| --- |
| **User Guide**  **Version 4.1.0** |

Table of Contents

[About 4](#_Toc430774456)

[Application 4](#_Toc430774457)

[License 4](#_Toc430774458)

[Terminology 4](#_Toc430774459)

[Pomodoro Technique®’s rules\* vs mAP 5](#_Toc430774460)

[Sub-tasking rules vs mAP 6](#_Toc430774461)

[Scrum’s practices\* vs mAP 6](#_Toc430774462)

[Configuration 7](#_Toc430774463)

[MySQL configuration 7](#_Toc430774464)

[Logger configuration 8](#_Toc430774465)

[Google Drive configuration 8](#_Toc430774466)

[NEW mAP start-up 9](#_Toc430774467)

[How to start mAP (3 ways) 9](#_Toc430774468)

[mAP settings 9](#_Toc430774469)

[How to set the preferences 9](#_Toc430774470)

[Getting Started 12](#_Toc430774471)

[The (Agile) Pomodoro technician at work 12](#_Toc430774472)

[The Scrum Master at work 13](#_Toc430774473)

[New task 13](#_Toc430774474)

[How to create tasks (5 ways) 13](#_Toc430774475)

[How to create subtasks (3 ways) 13](#_Toc430774476)

[How to use the Create panel 14](#_Toc430774477)

[Main keyboard shortcuts 14](#_Toc430774478)

[Activity List / Backlog 15](#_Toc430774479)

[How to read the main and sub-tables’ header 15](#_Toc430774480)

[Main table: How to use quick buttons and shortcuts 16](#_Toc430774481)

[Sub-table: How to use quick buttons and shortcuts 16](#_Toc430774482)

[How to use a task / subtask as template 16](#_Toc430774483)

[How to edit a task / subtask 17](#_Toc430774484)

[How to export data 17](#_Toc430774485)

[How to import data 18](#_Toc430774486)

[ToDo List / Iteration Backlog 18](#_Toc430774487)

[How to work with the timer 18](#_Toc430774488)

[How to read the main and sub tables’ header 20](#_Toc430774489)

[How to prioritize tasks 20](#_Toc430774490)

[Main table: How to use quick buttons and shortcuts 20](#_Toc430774491)

[Sub-table: How to use quick buttons and shortcuts 21](#_Toc430774492)

[How to use the timer 21](#_Toc430774493)

[How to use the timer’s toolbar 22](#_Toc430774494)

[How to merge tasks 23](#_Toc430774495)

[How to handle interruptions 23](#_Toc430774496)

[How to customize sounds 23](#_Toc430774497)

[Report List / Release Backlog 24](#_Toc430774498)

[How to read the Main table’s title bar 24](#_Toc430774499)

[How to read the Main and the Sub-tables’ headers 24](#_Toc430774500)

[Main table: How to use quick buttons and shortcuts 24](#_Toc430774501)

[Burndown / Burn-up Chart 25](#_Toc430774502)

[How to create charts 25](#_Toc430774503)

[How to create a Daily Burndown chart (Scrum) 26](#_Toc430774504)

[How to create a Release Product Burn-up chart (Scrum) 27](#_Toc430774505)

[Annex 27](#_Toc430774506)

[Themes 27](#_Toc430774507)

[Libraries 28](#_Toc430774508)

[XSD schema 28](#_Toc430774509)

# About

myAgilePomodoro (mAP) is a micro and macro time management software based upon Agile's best practices and the time management technique called the Pomodoro Technique®.

To make the most of mAP it is recommended to have knowledge of the Pomodoro Technique® (v1.3 and above) and Agile’s most common practices and artifacts (backlog management, iteration, story points...). Furthermore, mAP is "Agile methodology agnostic": to a certain extent, Scrum, RUP, XP… and even Kaban projects can be managed with mAP.

Originally called "myPomodoro", Phil Karoo took over the project starting with version 1.0.

Regarding the original versions and artworks, all credit goes to Paul Barton, Brian Wetzel, Jordan Smith and Robert Forscht.

# Application

mAP is a Java 6+, 32/64-bit, utf-8, multilingual, skinnable, standalone and portable application (no installer).

pomodoro16.png mAP supports utf-8 providing the font "Arial Unicode MS" is installed (usually bundled with Microsoft Windows, Office and Apple MAC OS X; see [Unicode ranges](http://www.microsoft.com/typography/fonts/font.aspx?FMID=1081) for language support). Otherwise, the system’s default font is used.

# License

mAP is a free open-source software, licensed under the [GNU General Public License version 3.0](http://www.gnu.org/licenses/gpl-3.0.txt) (GPLv3).  
All documentation and images are licensed under a [Creative Commons Attribution Non-commercial license](http://creativecommons.org/licenses/by-nc/4.0/).

# Terminology

* **Task**: any Activity, ToDo or Report item, regardless of the type.
* **Subtask**: any sub-Activity, sub-ToDo or sub-Report item, regardless of the type.
* **Main** **table**: table of tasks.
* **Sub**-**table**: table of subtasks.
* **Parent** **task**: task that has subtasks.
* **Started** **task** / **subtask**: task / subtask with, at least, one real pomodoro.
* **Running** **task** / **subtask**: task / subtask with a running pomodoro.
  + Running pomodoro: period of time (pomodoro) when the timer is started or paused (running).
* **Finished** **task** / **subtask**: **started** task / subtask with as many real pomodoros as the sum of its estimated and overestimated pomodoros.
* **Working set**: length of time defined by the number of pomodoros per set (see [**How to set the preferences**](#HowtosetthepreferencesWorkingSet)).
* **Working** **day**: length of time defined by the max number of pomodoros per day (see [**How to set the preferences**](#HowtosetthepreferencesWorkingDay)).

# Pomodoro Technique®’s rules\* vs mAP

\* as per the official book of the Pomodoro Technique®.

mAP fully implements the Pomodoro Technique®’s specification (v1.3 and above). Therefore mAP can be used to strictly follow the rules of the technique.

However, mAP is flexible in many ways. In fact, mAP has features that are not "permitted" or covered by the specification (eg sub-tasking). There are hereunder highlighted in orange.

|  |  |
| --- | --- |
| **Rule** | **mAP** |
| A Pomodoro Consists of 25 minutes Plus a Five-Minute Break | Configurable in **Preferences**.  mAP allows shortening / lengthening / pausing pomodoros and short breaks.  mAP allows stopping automatically the timer at the end of pomodoros using the timer’s workflow interruption button. |
| After Every Four Pomodoros Comes a 15-30 Minute Break | Configurable in **Preferences**.  mAP allows shortening / lengthening / pausing long breaks.  mAP allows restarting a Set after stopping the timer. |
| The Pomodoro Is Indivisible. There are no half or quarter Pomodoros. | mAP allows shortening pomodoros. |
| If a Pomodoro Begins, It Has to Ring:   * If a Pomodoro is interrupted definitively – i.e. the interruption isn’t handled – it’s considered void, never begun, and it can’t be recorded with an X. * If an activity is completed once a Pomodoro has already begun, continue reviewing the same activity until the Pomodoro rings. | Ringing and ticking sounds are customizable (see [**How to customize sounds**](#Howtocustomizesounds)) and possible to disable in **Preferences** or using the timer’s mute button.   * mAP allows creating unplanned tasks and subtasks nonetheless. * mAP allows shortening pomodoros. |
| Protect the Pomodoro. Inform effectively, negotiate quickly to reschedule the interruption, call back the person who interrupted you as agreed. | mAP allows creating unplanned tasks and subtasks at any time and internal and external interruption tasks and subtasks during pomodoros. |
| If It Lasts More Than 5-7 Pomodoros, Break It Down. Complex activities should be divided into several activities. | mAP allows duplicating tasks and creating subtasks. |
| If It Lasts Less Than One Pomodoro, Add It Up. Simple tasks can be combined. | mAP allows merging tasks and subtasks.  mAP allows starting tasks and subtasks with no estimation (0 pomodoro). |
| Results Are Achieved Pomodoro after Pomodoro. | mAP measures accuracy and analyses errors (diff).  mAP allows creating burn-up and burndown charts. |
| The Next Pomodoro Will Go Better. | Just stick to mAP ;-) |

# Sub-tasking rules vs mAP

mAP implements two new rules to support sub-tasking. We suggest these rules be added to the Pomodoro Technique®’s specification.

|  |  |
| --- | --- |
| **Rule** | **mAP** |
| All the rules of the Pomodoro Technique® apply to sub-activities. | mAP allows creating, estimating and running subtasks. |
| Activities with sub-activities shouldn’t be estimated. Only sub-activities should. | mAP goes one step forward: it allows estimating and running parent tasks. Any change of pomodoros in a subtask is equally applied to its parent task. |

# Scrum’s practices\* vs mAP

\* practices, artifacts, indicators...

|  |  |
| --- | --- |
| **Practice** | **mAP** |
| Backlogs | mAP allows [ordering](https://www.scrumalliance.org/community/articles/2011/august/it%E2%80%99s-ordered-%E2%80%94-not-prioritized!) the **Backlog** (Product Backlog), prioritizing the **Iteration Backlog** (Sprint Backlog) and managing the **Release Backlog**. |
| Meetings | mAP may be used to monitor Sprint planning, Daily Scrum, Sprint review and Sprint retrospective meetings with the Pomodoro Technique® (having breaks during meetings is not a bad idea).  Simply create a task with a made-up **type** and set the **estimate** (time boxing). In the preferences, set the **length** of pomodoros and breaks that fits the meeting best. |
| Items | mAP allows creating **tasks** and **subtasks** (items) with a set of pre-defined **types** (User Story, Epic…).  For convenience, tasks with type "User Story" present a skeleton of story ("As a {role}, I can…"). |
| Sprints | mAP allows setting the **Iteration** (Sprint) number on the tasks.  mAP allows iteration with number 0 for project ramp-up, initial technical stories, spikes…  mAP allows moving tasks from the **Iteration Backlog** back to the **Backlog** (descoping)**.**  mAP allows moving tasks from the **Iteration Backlog** (In progress) to the **Release Backlog** (Done)**.**  mAP allows re-opening tasks from the **Release Backlog** back to the **Backlog** (not Done-Done).  mAP allows prioritizing tasks AND subtasks in the **Iteration Backlog**. |
| Indicators | mAP shows on the backlogs’ title bar the sum of story points (**SP**) of the tasks. For convenience, on the **Release Backlog**’s title bar**,** and when tasks are selected, it reads "**V**" (for Velocity).  mAP also shows the **P**roductivity (story points per day) on the **Release Backlog**’s title bar when tasks are selected. |
| Charts | mAP allows creating burn-up, burn-down charts with target, guide and scope lines upon different configurable set of tasks.  A Scrum Master can easily create a Daily Burndown chart as a Product Owner can issue a Release Product Burn-up chart based on dates or a range of iterations. |

# Configuration

mAP allows two different usages:

* **Local database** (SQLite) (default): this is meant to be used by individuals. At first start-up, mAP will create its own local database file (myagilepomodoro.db).
* **Remote database** (MySQL): this is meant to be used by co-located small to medium size teams (see [**MySQL configuration**](#MySQLconfiguration)). However, an individual may want to use a local MySQL server to make mAP quicker at handling data.

mAP allows two different modes:

* **Agile Mode** (default): Agile + Pomodoro modes.
* **Pomodoro Mode**: only knowledge of the Pomodoro Technique® is required to properly manage tasks and reports.

pomodoro16.png Decide first for the **usage** then for the **mode** (see [**mAP configuration**](#mAPConfiguration)).

## MySQL configuration

1. It is recommended to configure the server to support utf-8 characters. For that, utf-8 encoding must be enabled in MySQL’s my.ini file:

|  |
| --- |
| collation\_server=utf8\_unicode\_ci  character\_set\_server=utf8  skip-character-set-client-handshake |

1. Database <database name> must be created.
   * If the server is not meant to support utf-8 characters, the database <database name> may be configured to support utf-8 by itself. Refer to MySQL’s documentation.
2. A text file called "mysql.properties" must be added to the directory where mAP is located:

|  |
| --- |
| database=<database name>  host=<database server address / IP>:<port > (default port number: 3306)  user=<user name>  password=<password> (may be empty) |

Note: the account (user/password) must have the permission to create tables in the database <database name>.

1. Finally, mAP must be run once to automatically create the tables it needs in the database <database name>.

After the tables are created, each team member must also create her/his own properties file with her/his own credentials to connect to the database <database name>.

pomodoro16.png It is recommended to grant Write, Update and Delete permissions on table "Preferences" only to the person in charge of configuring the settings (steps 1 to 4). Likewise, granting Delete permission on table "Activities" to that person only may prevent lost of data.

## Logger configuration

At first start-up, mAP will create an error log file (myAgilePomodoro.log). Most of the time, when an error happens, mAP will write the trace down into the file.

pomodoro16.png mAP must be given the privilege to read and write on the local file system.

## Google Drive configuration

To be able to export data to Google Drive, a text file called "google.properties" must be created in the directory where mAP is located:

|  |
| --- |
| clientid=<client ID>  clientsecret=<client secret>  redirecturi=<redirect URI>  (see [Google Auth](https://developers.google.com/drive/web/about-auth)) |

Example:

|  |
| --- |
| clientid=289346736832-fjowijfiowe09rqwrwyhrduffjsid.apps.googleusercontent.com  clientsecret=mVtt-Bq4y672MCoJk88fB900  redirecturi=urn:ietf:wg:oauth:2.0:oob |

At start-up time, mAP will add the "Google Drive" format to the **File Format** field of the export form.

# NEW mAP start-up

## How to start mAP (3 ways)

1. Double-click on the executable (.exe for Windows; .jar for other systems)
2. Run on a command line
   * java -jar "<full path to the executable .exe or jar>"
     + Example: java -jar "C:\map\myAgilePomodoro.jar"
3. Run on a command line with option to create / use database, config files and log file on a different path from the executable
   * java -jar "<full path to the executable .exe or jar>" "<full path to the database file, config files and log file, without file separator at the end>"
     + Example: java -jar "C:\map\myAgilePomodoro.jar" "C:\map\data"

pomodoro16.png If the executable’s path or the optional path contains some "special" characters, mAP may not be able to start up (refer to the error log file for details).

# mAP settings

## How to set the preferences

File > Preferences

* Modes
  + **Agile Mode**: Agile + Pomodoro.

It is an Agile’s common practice for tasks to last two days at maximum which is more or less equivalent to 20 pomodoros (**Max nb pom / Task**) of 25 minutes (**Pomodoro length**). However it is a Pomodoro Technique®’s sub-tasking rule for subtasks to last 5 pomodoros of 25 minutes (**Pomodoro length**) at maximum (see [Sub-tasking rules vs mAP](#SubtaskingrulesvsmAP)).

* + **Pomodoro Mode**: Pomodoro only.

It is a Pomodoro Technique®’s rule for tasks and subtasks to last 5 pomodoros (**Max nb pom / Task**) of 25 minutes (**Pomodoro length**) at maximum.

* **Pomodoro length**: length of pomodoros (minutes). Used to set the timer.
* **Short break length**: length of short breaks (minutes). Used to set the timer.
* **Long break length**: length of long breaks (minutes). Used to set the timer.
* **Max nb pom / Day**: maximum number of pomodoros per day. Define the length of a **working** **day** depending on the mode of time calculation (see below Time Calculation).
  + (**Pomodoro** **Mode**) Also used to prevent the user from adding too many pomodoros to the **ToDo List** (the Pomodoro Technique® states that the "ToDo Today Sheet" is meant for one day of work at a time).
  + In bracket, the length of a working Day is displayed in hours.
* **Nb pom / Set**: number of pomodoros per set. Define the length of a **working** **set** depending on the mode of time calculation (see below Time Calculation).
  + Also used to set the timer. Long breaks happen every set of pomodoros.
  + In bracket, the length of a working Set is calculated after the length of the working day and displayed in days / hours.
* **Max nb pom / Task**: maximum number of pomodoros per task / subtask.
  + Used to prevent the user from creating tasks with too many pomodoros.
  + In bracket, the max length of tasks is calculated after the length of the working day and displayed in days / hours.
* Time calculation
  + **Plain Hours** (default): used to convert estimates (pomodoros + breaks) into **working** days and sets ("<days> days <hours> : <minutes>").

|  |  |
| --- | --- |
|  | **Example of preferences** |
| Pomodoro length | 25 min |
| Short break length | 5 min |
| Long break length | 20 min |
| Nb pom / Set | 4 pomodoros |
| Plain work per set | 4 pomodoros  + 3 short breaks  + 1 long break |
| **Working set** | **2 hours : 15 minutes** |
| Max nb pom / Day | 10 pomodoros |
| Plain work per day | 10 pomodoros  + 8 short breaks  + 2 long breaks |
| **Working day** | **5 hours : 30 minutes** |

Preferences applied to estimates:

|  |  |  |
| --- | --- | --- |
|  | **Example 1** | **Example 2** |
| Estimate | 2 pomodoros | 12 pomodoros |
| Plain work | 2 pomodoros  + 2 short breaks | 1 (working) day  + 2 pomodoros  + 2 short breaks |
| **Time** | **1 hour** | **1 (working) day**  **1 hour : 00 minute** |

* + **Effective Hours**: used to convert estimates (pomodoros only) into **working** days and sets (“<days> days <hours> : <minutes>”).

|  |  |
| --- | --- |
|  | **Example of preferences** |
| Pomodoro length | 25 min |
| Nb pom / Set | 4 pomodoros |
| Effective work per set | 4 pomodoros |
| **Working set** | **1 hour : 40 minutes** |
| Max nb pom / Day | 10 pomodoros |
| Effective work per day | 10 pomodoros |
| **Working day** | **4 hours : 10 minutes** |

Preferences applied to estimates:

|  |  |  |
| --- | --- | --- |
|  | **Example 1** | **Example 2** |
| Estimate | 2 pomodoros | 12 pomodoros |
| Effective work | 2 pomodoros | 1 (working) day  + 2 pomodoros |
| **Time** | **50 minutes** | **1 (working) day**  **0 hour : 50 minutes** |

* **Language**: language of the application. Used to format dates and time.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **English** | **English US** | **Other** |
| Date | dd MMM yyyy | MMM dd yyyy | dd MMM yyyy |
| Example | 06 Sep 2011 | Sep 06 2011 | 06 Sep 2011 |
| Time | hh:mm a | hh:mm a | HH:mm |
| Example | 04:35 PM | 04:35 PM | 16:35 |

* Sounds
  + **Ticking**: enable/disable ticking. Used to set the timer. Ticking happens during pomodoros only.
  + **Ringing**: enable/disable ringing. Used to set the timer. Ringing happens at the end of pomodoros and the start of pomodoros after the break.
* System Tray (appears for OS that support the System Tray feature only)
  + **System Tray**: enable/disable system tray icon.
  + **Pop-up message**: enable/disable system tray popup messages.
* Window behavior
  + **Always On Top**: set mAP to remain on top of others applications.
  + **Bring To Front**: set mAP to automatically come to the front of others applications when a pomodoro starts (ring), ends (ring) or a task finishes. Likely use case: mAP is minimized in the System tray or hidden in the background and the sound is muted.
* Themes: mAP‘s look and feel.

Press ENTER or the Save button to save.

# Getting Started

mAP makes the management of tasks a simple process. Only timing differs: Agile projects take weeks/months whereas the management of ToDo lists is often a matter of days.

## The (Agile) Pomodoro technician at work

1. The first time, mAP opens up on the **Preferences** panel: choose the mode of work, **Agile** (default) or **Pomodoro**. Save and restart if needed.
   * Basically, choosing Agile means estimating tasks with story points on top of the estimation with pomodoros.
2. Open the **Create Panel** and create tasks. Tasks are automatically added to the **Activity List** / **Backlog** (also known as "Activity Inventory Sheet" in the Pomodoro Technique®).
   * As tasks can be duplicated, tasks may be used as templates.
   * Creating test data is another way to get started with mAP.
3. Open the **Activity List** / **Backlog** panel
   * Add subtasks to tasks if necessary. As subtasks can be duplicated, subtasks may be used as templates.
   * Estimate tasks and subtasks.
4. Move (>>) tasks to the **ToDo List** / **Iteration Backlog** (also known as "ToDo Today Sheet" in the Pomodoro Technique®).
5. Open the **ToDo List** / **Iteration Backlog** panel.
   * Estimate tasks and subtasks.
   * Write comments (**Pomodoro Mode**) / stories (**Agile Mode**).
   * As Product Owner, prioritize the tasks and subtasks of the **ToDo List** / **Iteration Backlog**.
   * Remove (<<) tasks considered out of scope.
6. Work on tasks and subtasks using (or not using) the timer / the Pomodoro Technique®.
   * Advanced: one can work on a parent task and its subtasks independently. As a result, the parent task may end up having more real pomodoros than the sum of the real pomodoros of its subtasks.

Likely use case: do some work on a parent task that doesn’t worth creating a subtask for.

1. Complete (>>) (**Pomodoro Mode**) / Set as "Done" (**Agile Mode**) tasks in the **Report List** / **Release Backlog** (also known as "Records Sheet" in the Pomodoro Technique®).
2. Open the **Report List** / **Release Backlog** panel
   * "Reopen" tasks considered as not completed (**Pomodoro Mode**) / not "Done Done" (**Agile Mode**).
3. Issue **Burndown** and **Burn-up** charts.

## The Scrum Master at work

1. At the beginning of the Sprint, timebox the Sprint planning meeting with pomodoros
   1. In the preferences, set the **length** of pomodoros and breaks that fits the meeting best. Switch off the ticking. Choose **Pomodoro Mode** if the meeting needs to be scheduled. Save and restart.
   2. Create a task with a made-up **type** and set the **estimate**.

(**Pomodoro Mode**) Set a scheduled date.

* 1. Move the task to the **ToDo List** / **Iteration Backlog** and start.
     + Create unplanned meeting and interruption tasks if necessary.
  2. At the end of the meeting, revert the changes.

1. During the Sprint:
   1. Issue the Daily Burndown chart of the day (see [**How to create a Daily Burndown chart**](#HowtocreateaDailyBurndownchart)) and indicators (Sprint velocity and productivity).
   2. Timebox the Daily Scrum meeting.
   3. Create and work on specific tasks regarding Scrum Master’s "duties" such as
      * Help improve the ordering of the **Product Backlog**.
      * Go through the **Product Backlog** and **Iteration Backlog** to review epics and stories (story line, acceptance criteria…).
      * Go through the **Iteration Backlog** to identify impediments.
2. At the end of the Sprint:
   1. Issue the Sprint Burndown chart and indicators.
   2. Timebox the Sprint review and retrospective meetings.

# createButton2.pngNew task

File > New Task

## How to create tasks (5 ways)

1. Use the Create panel.
2. Use the quick buttons (createcreate.png, duplicateduplicate.png, unplannedunplanned.png, interruptionsexternal.pnginternal.png ).
3. Use shortcuts (create, duplicate, unplanned, interruptions)
4. Create test data
   1. Data > Generate Test Data
5. Import data.

## How to create subtasks (3 ways)

1. Use the quick buttons (createcreate.png, duplicateduplicate.png, unplannedunplanned.png, interruptionsexternal.pnginternal.png ).
2. Use shortcuts (create, duplicate, unplanned, interruptions)
3. Create test data
   1. Data > Generate Test Data *with subtasks*

## How to use the Create panel

* (**Pomodoro Mode**) **Date scheduled**: start date. This field is mandatory.
* **Title**: name of task. This field is mandatory.
* **Estimated Pomodoros**: estimated number of pomodoros (the Pomodoro Technique® calls this "First estimate").
* (**Agile Mode**) **Story Points**
* (**Agile Mode**) **Iteration**
* **Type**: type of task. This editable list is pre-filled with existing types. When creating / editing a task, the list is updated. Unused types are removed at start up time. The list is also pre-filled with some common Agile "types" such as "User Story" (**Agile Mode**).
* **Author**: name of the author. This editable list is pre-filled with existing authors. When creating / editing a task, the list updated. Unused authors are removed at start up time.
* **Place**: place where the task takes places. This editable list is pre-filled with existing places. When creating / editing a task, the list is updated. Unused places are removed at start up time.
* **Description**: description of the task in case the title isn’t self-explanatory. HTML 3.2 may be used.

Press ENTER in the Type text field or the Save button to save.

# Main keyboard shortcuts

* "File" and "View" menus shortcuts:
  + ESCAPE: exit mAP.
  + ALT + C: display the **Create** panel.
  + ALT + S: display the **Splash** screen.
  + ALT + P: display the **Preferences** panel.
  + ALT + A: display the **Activity List** / **Backlog** panel.
  + ALT + T: display the **ToDo List** / **Iteration Backlog** panel.
  + ALT + R: display the **Report List** / **Release Backlog** panel.
  + ALT + B: display the **Charts** panel.
* Preferences and Create panel shortcut:
  + ENTER: save the form.
* All **List / Backlogs** shortcuts:
  + ALT + M: maximize the window / show the original window to previous size and location.
    - **ToDo List** / **Iteration Backlog**: maximize the window / show the timer only / show the timer and the list / show the original window to its previous size and location.
  + CTRL + A: select all tasks / subtasks.
  + CTRL + F: search task.
  + CTRL + <tab number> (starting from 1): open corresponding tab.
  + F2: edit the selected task / subtask’s title.
* **Comment / Story** panel shortcuts:
  + **Preview**, **Editor** and **HTML** views
    - CTRL + A: select all text.
    - CTRL + S: save.
    - CTRL + C: copy selected text.
    - CTRL + V: paste without formatting any string content found in the clipboard.
  + **Preview** and **Editor** views
    - CTRL + B: turn selected text bold.
    - CTRL + I: turn selected text italic.
    - CTRL + U: underline selected text.
    - CTRL + L: insert unordered list at selected text’s position.
    - CTRL + O: insert ordered list at selected text’s position.
  + **Editor** view
    - CTRL + L: insert unordered list at caret position.
    - CTRL + O: insert ordered list at caret position.
    - CTRL + R: remove list item at caret position.

pomodoro16.png The action of some buttons (eg Save) may be triggered by shortcuts. In such case, the name or the combination of the keyboard keys appears in upper case when hovering over the button.

# createButton2.pngActivity List / Backlog

View > Activities / Backlog

By default, the main table and the sub-table are sorted by scheduled date (**Pomodoro Mode**) / iteration and title (**Agile Mode**).

## How to read the main and sub-tables’ header

* (**Pomodoro Mode**) **U**: "✔" if the task is unplanned or is an interruption.
* (**Pomodoro Mode**) **Date**: start date. The date is strikenthrough when the task is overdue.
* **Title**: name of task. This column is editable.
* **Type**: type of task. This column is editable.
* **Estimated**: estimated (+ overestimated) pomodoros.
* (**Agile Mode**) (Main table) **SP**: Story Points.
* (**Agile Mode**) (Main table) **IT**: Iteration.

## Main table: How to use quick buttons and shortcuts

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Shortcut** | **Action** | **Condition** |
| selected.png | **CTRL + G** | Scroll to the selected task(s) and selected subtask(s) if any. |  |
| duplicate.png | **CTRL + D** | Duplicate the selected task with "(D)" added to the title, overestimated and real pomodoros reset to 0 and date set to today. |  |
| create.png | **CTRL + T** | Create a task with default title "(N) New task", estimated pomodoro set to 0 and date set to today. |  |
| refresh.png | - | Update the list from the database. | **Remote database** |
| - | **SHIFT**  **+ '>'** | Move the selected task(s) to **ToDo List** / **Iteration Backlog** (same as button ">>"). |  |
| - | **DEL** | Delete the selected task(s) and their subtask(s) (same as button "Delete"). |  |

## Sub-table: How to use quick buttons and shortcuts

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Shortcut** | **Action** | **Condition** |
| duplicate.png | **CTRL + D** | Same as above. |  |
| create.png | **CTRL + T** | Same as above. |  |
| - | **DEL** | Same as above.  The pomodoros of the selected subtask(s) are subtracted from the pomodoros of the parent task. |  |

## How to use a task / subtask as template

1. Select the task / subtask to use as template.
2. Press CTRL + D: a duplicate of the selected task / subtask is created at the end of the list. Keep pressing CTRL + D to create as many copies as necessary.

## How to edit a task / subtask

1. Select the task / subtask to edit.
2. (**Remote database**) refresh.png Update the list to make sure you have the latest version of the task / subtask.
3. Change the details:
   * In-line: Select a cell and edit the value (not all columns are editable).
   * "Edit" tab: edit the details. Press the Save button to save.

## How to export data

1. Select the tasks (all file formats) or subtasks (all file formats except **XML**) to export.
2. Open the Export tab.

* **Header** (all file formats except **XML**): first row / header of the file (column names).
* **File name**: name of the export file. 
  + If left empty, default name "myAgilePomodoro" is used
  + If it contains some special characters not supported by the local file system, the export will fail.
* **File format**: type of formatting. This field is mandatory.

The **XML** file format is the only file format that allows exporting and importing tasks and subtasks alongside. Other formats allow exporting and importing either tasks or subtasks.

|  |  |  |
| --- | --- | --- |
|  | **Export** | **Import** |
| **XML** | Tasks and subtasks | Tasks and subtasks |
| **Other formats** | Tasks or subtasks | **NEW** Tasks or subtasks |

* **Date pattern** (**CSV**, **XML** and **Google Drive** file formats only): sets of patterns for days, months and years. These fields are mandatory.

|  |  |  |
| --- | --- | --- |
|  | **Pattern** | **Example** |
| **Day** | D | 1 |
| Dd | 01 |
| **Month** | M | 1 |
| MM | 01 |
| MMM | Jan |
| MMM | January |
| **Year** | yy | 01 |
| yyyy | 2001 |

Example: M/dd/yyyy 🡪 1/01/2001

* **Separator** (**CSV** and **Google Drive** file formats only): set of pre-defined separators, plus an empty editable field (this must be a character, not a string). This field is mandatory (if this field is left empty or not a proper character, default separator "Comma" is used).

1. Press the Export button to export to the local file system. If the export fails, mAP will write the trace down into the error log file.

pomodoro16.png mAP creates a file with name "<*File name*>" and extension ".<*File format*>" in the directory where mAP is located.

pomodoro16.png Only raw text and line breaks are exported (no formatting).

waiting.jpg The export feature is one of the few features that use the waiting cursor (import, move, complete, prioritize, delete, test data…). Beware that none of these features will start if one of them is currently running.

## How to import data

As import and export files have the exact same format, to get started, simply export some dummy data. Then use the exported file as template. When importing, make sure to select the type of task (**NEW Tasks** or **Subtasks**), the **File format,** the **Date pattern** and the **Separator** for mAP to parse the import file.

 XML files are checked for validation against a XSD schema file (see [XSD schema](#XSDSchema)).

 Import from Google Drive hasn’t been implemented.

# createButton2.pngToDo List / Iteration Backlog

View > ToDos / Iteration

By default, the main table and the sub-table are sorted by priority.

## How to work with the timer

1. Select a task or a subtask.
2. Start the timer
   * If the task / subtask is finished 🡪 Step 1 or Overestimate it 🡪 Step 2.
   * If a task / subtask was previously voided or a short break stopped, decide whether to restart a new Set or not.
     + Restart a Set after an unusually long break such as a lunch or a meeting.
     + Do not restart the Set after an urgent interruption.

(**Remote database**) By the time, if the task / subtask has been changed by someone else, mAP will ask you to update the list (refresh.png).

1. Work on the task / subtask
   * Hover over the timer to know when the next break happens.
   * If the selected task / subtask is voided 🡪 Step 1.
   * If an urgent internal or external interruption task or subtask is created, stop the timer and process the urgent task / subtask 🡪 Step 1.

Before the end of the pomodoro, a different task / subtask may be selected for the next pomodoro to come 🡪 Step 1 then Step 4.

1. Wait for the timer to ring (end of pomodoro)
   * If the workflow has been stopped 🡪 Step 2.

(**Remote database**) By the time, if the task / subtask has been changed by someone else, mAP will automatically update it.

(**Remote database**) By the time, if the task / subtask has been finished by someone else, mAP will automatically overestimate it by 1 to record the pomodoro.

1. Take a break
   * If the break is stopped 🡪 Step 1 or Step 2.
   * If working on a different task / subtask is needed 🡪 Step 1 then Step 6.
2. Wait for the timer to ring (end of break)
   * If the selected task / subtask is finished the timer stops by itself 🡪 Step 1 or Overestimate it 🡪 Step 2.
   * 🡪 Step 3.

(**Remote database**) By the time, if the task / subtask has been changed by someone else, mAP will automatically update it.

pomodoro16.png As per the Pomodoro Technique®, icons are used to show pomodoros and interruptions:

* Pomodoro: square (not yet done)
* Real Pomodoro: squareCross (done)
* Internal interruption: quote
* External interruption: dash

pomodoro16.png To keep users honest about the Pomodoro Technique®, mAP doesn’t allow re-estimating **started** tasks and subtasks.

## How to read the main and sub tables’ header

* **Priority**: row number
* (**Pomodoro Mode**) **U**: "✔" if the task is unplanned or is an interruption.
* **Title**: name of task. This column is editable.
* **Estimated**: number of real / estimated (+ overestimated) pomodoros. This number can be decreased / increased as long as the task hasn’t **started**.
* (**Agile Mode**) (Main table) **SP**: Story Points.
* (**Agile Mode**) (Main table) **IT**: Iteration.

## How to prioritize tasks

1. Drag and drop task(s).
2. Wait for the progress bar to be “Done”

## Main table: How to use quick buttons and shortcuts

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Shortcut** | **Action** | **Condition** |
| start.png | - | Foldstart.png / Expandstart.png the quick buttons toolbar. |  |
| selected.png | **CTRL + G** | Scroll to the selected task(s), and selected subtask(s) if any. |  |
| running.png | **CTRL + G** | Scroll to and select the running task, and subtask if any. |  |
| plusone.png | **-** | Overestimate the selected task by one pomodoro. | The selected task must be finished or already overestimated. |
| unplanned.png | **CTRL + U** | Create an unplanned task with default title "(U) Unplanned", estimated pomodoro set to 0 and date set to today. This may not work when the **Comment / Story** panel is in **Editor** or **HTML** mode. |  |
| external.png | **CTRL + E** | Create an external interruption with default title "(E) External", estimated pomodoro set to 0 and date set to today.  This may not work when the **Comment / Story** panel is in **Editor** or **HTML** mode. | A pomodoro must be running. |
| internal.png | **CTRL + I** | Create an internal interruption with default title "(I) Internal", estimated pomodoro set to 0 and date set to today.  This may not work when the **Comment / Story** panel is in **Editor** or **HTML** mode. | A pomodoro must be running. |
| refresh.png |  | Update the list from the database. | **Remote database** |
| - | **SHIFT**  **+ '<'** | Move the selected non-running task(s) back to **Activity List** / **Backlog** (same as button "<<"). The priorities of the subtasks are reset. |  |
| - | **SHIFT**  **+ '>'** | Move the selected non-running task(s) to **Report List** / **Release Backlog** (same as button "Complete / Done"). |  |

## Sub-table: How to use quick buttons and shortcuts

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Shortcut** | **Action** | **Condition** |
| start.png | **-** | Same as above. |  |
| duplicate.png | **CTRL + D** | Duplicate the selected subtask with "(D)" added to the title, overestimated and real pomodoros reset to 0 and date set to today. |  |
| plusone.png | **-** | Same as above. | Same as above. |
| create.png | **CTRL + T** | Create a subtask with default title "(N) New subtask", estimated pomodoro set to 0 and date set to today. |  |
| unplanned.png | **CTRL + U** | Same as above. |  |
| external.png | **CTRL + E** | Same as above. | Same as above. |
| internal.png | **CTRL + I** | Same as above. | Same as above. |
| - | **DEL** | Delete the selected non-running subtask(s) (same as button "Delete").  The pomodoros of the selected non-running subtask(s) are subtracted from the pomodoros of the parent task. |  |

## How to use the timer

|  |  |  |
| --- | --- | --- |
|  | **Action** | **Condition** |
| start.png | Start a pomodoro. | The selected task / subtask must not be finished. |
| stop.png | Void a pomodoro or stop a break. | The timer must be started or paused. |
| pause.png | Pause a pomodoro or a break. | The timer must be started. |
| resume.png | Resume a pomodoro or a break. | The timer must be paused. |
| timeplus.png | Lengthen a pomodoro or a break. |  |
| timeminus.png | Shorten a pomodoro or a break. |  |

## How to use the timer’s toolbar

Depending on the theme, clicked buttons are flattened and/or grayed out, or simply repainted with a different icon.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **State** | **Action** | **Details** |
| continuous.png | Workflow ON | Set the timer to stop at the end of pomodoros (workflow interruption). | Pomodoros that follow must be started manually. |
| sound.png | Sound ON | Mute the ticking and the ringing. | Ringing happens at the end of pomodoros and the start of pomodoros right after the break. There is no ticking during breaks.  This won’t appear if the permanent setting **Ticking** or **Ringing** is disabled in **Preferences**. |
| unpin.png | Pin OFF | Set mAP to remain on top of others applications. | This can also be used to temporarily disable the permanent setting **Always on top** in **Preferences**. |
| upsize.png | Not maximized | Resize the window. | This resizes the window as follows: max, **NEW** small timer\*, timer, timer + list, original window set to previous size and location.  mAP will keep track of the button’s location and move the cursor accordingly. |

\* **NEW** Small size timer:

Capture.JPG

## How to merge tasks

1. Select the tasks or subtasks to merge (excluding currently running task / subtask).
2. Open the Merging tab, set the details of the new task / subtask and save.
3. As a result, the selected tasks / subtasks are deleted and the new task / subtask is added to the **ToDo List / Iteration Backlog** main table / sub-table.
   * Real, estimated and overestimated pomodoros are summed up.
   * Comments / Stories are aggregated with reference to their original task.
   * Subtasks are aggregated if any.

(**Pomodoro Mode**) If the start date of the new task isn’t today the new task is added to the **Activity List**.

pomodoro16.png (**Remote database**) If the user has no delete permission on table "Activities", although the selected task will disappear from the table it won’t be deleted from the database and will reappear at next start-up.

## How to handle interruptions

1. Select a task / subtask.
2. Start the timer.
3. Work on the task / subtask.
4. Whenever an interruption happens, create an interruption using the appropriate quick buttons, shortcuts (internal or external) or open the Unplanned / Interruption tab, set the details of the new task and save.
   * Type: internal interruption or external interruption.
   * (**Pomodoro Mode**) **Date scheduled**: start date. If the task is urgent set the date to today. This field is mandatory.
5. If the interruption is urgent: void the pomodoro of the running task / subtask (stop the timer), select the interruption and start it straight away (no break).

pomodoro16.png Unplanned tasks aren’t interruptions but interruptions are unplanned tasks. (**Pomodoro Mode**) If the start date is today they are added to the **ToDo List** otherwise to the **Activity List**.

## How to customize sounds

* Ticking: create a .wav file called "ticking.wav" in the directory where mAP is located. Make it short as mAP will loop over it.
* Ringing: create a .wav file called "ringing.wav" in the directory where mAP is located. mAP will play it once at the end of the breaks and at the start of a pomodoro after a break.

pomodoro16.png mAP must be given the privilege to read files on the local file system.

# createButton2.pngReport List / Release Backlog

View > Reports / Done

By default, the main table is sorted by date of completion / done. However the sub-table is sorted by priority.

## How to read the Main table’s title bar

* **Report List / Release Backlog**: (number of selected tasks /) number of tasks.
* **Done**: number of real / estimated (+ overestimated) pomodoros
* **A** (Accuracy): global success rate = *real / (estimated + overestimated) \* 100*. 100% means that all pomodoros have been done.
* (**Agile Mode**) **SP** (Story Points)/ **V**elocity: sum of story points of all tasks / selected tasks.
* **P** (Productivity): number of pomodoros (**Pomodoro Mode**) / story points (**Agile Mode**) per day for the selected tasks.
* Quick buttons

## How to read the Main and the Sub-tables’ headers

* (**Pomodoro Mode**) **U**: "✔" if the task is unplanned or is an interruption.
* (Main table) **Date**: date of completion.
* **Title**: name of task. This column is editable.
* **Type**: type of task.
* **Real**: number of real / estimated (+ overestimated) pomodoros.
* **D I** (Diff I): difference between the number of real and estimated pomodoros (Diff I = *real - estimated*).
* **D II** (Diff II): difference between the number of real and estimated and overestimated pomodoros (Diff II = *real -* *estimated - overestimated*). This value is displayed only if there are overestimated pomodoros.
* (**Agile Mode**) (Main table) **SP**: Story Points.
* (**Agile Mode**) (Main table) **IT:** Iteration.

## Main table: How to use quick buttons and shortcuts

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Shortcut** | **Action** | **Condition** |
| selected.png | **CTRL + G** | Scroll to the selected task(s) and selected subtask(s) if any. |  |
| refresh.png | - | Update the list from the database. | **Remote database** |
| - | **SHIFT**  **+ '<'** | Reopen / Move the selected task(s) back to **Activity List** / **Backlog** with "(R)" added to the title (same as button "Reopen"). The priorities of the subtasks are reset. |  |
| - | **DEL** | Delete the selected task(s) and their subtask(s) (same as button "Delete"). |  |

# createButton2.pngBurndown / Burn-up Chart

View > Charts

## How to create charts

1. Choose **Burndown** and / or **Burn-up Chart**. If the Scope option is selected, only the Burn-up Chart option is selectable.
   * **Burndown Chart**:
     + Type: type of data (**Story Points**, **Pomodoros**…)
     + **%**: remaining data displayed in percentage of the initial / maximum value of the target line.
     + **Y-Legend**: legend on the left of the chart. May be empty.
     + **X-Legend**: legend on the bottom of the chart. May be empty.
     + **Color**: color of the chart. Press to change.
     + **Target**:
       - **X-Legend**: legend on the bottom of the chart. May be empty.
       - **Color**: color of the target line. Press to change.
   * **Burn-up Chart**:
     + Type: type of data (**Story Points**, **Pomodoros**…)
     + **%**: data (of tasks completed / done) displayed in percentage of the last / maximum value of the scope line.
     + **Y-Legend**: legend on the left of the chart (‘right’ if "Burndown Chart" is selected). May be empty.
     + **X-Legend**: legend on the bottom of the chart. May be empty.
     + **Color**: color of the chart. Press to change.
     + **Guide**:
       - **X-Legend**: legend on the bottom of the chart. May be empty.
       - **Color**: color of the guide line. Press to change.
     + **Scope**:
       - **X-Legend**: legend on the bottom of the chart. May be empty.
       - **Color**: color of the scope line. Press to change.
2. Configure the type of **Tasks** and the **Dimensions** of the chart.
   * **Tasks**
     + Type:
       - **ToDo List** / **Iteration Backlog + Report List** / **Release Backlog**: tasks of both backlogs.
       - **Report List** / **Release Backlog**: tasks of the **Report List** / **Release backlog** only. This excludes tasks not yet completed / done. This option is only relevant to **burn-down** charts.
       - (**Agile Mode**) **Iteration**: tasks of a specific iteration.
     + **Dates**:
       - **Dates**: start and end dates as range for the X-axis of the chart. If the length exceeds 2 workings weeks (10 days), only Mondays are displayed. If it exceeds 5 working months (100 days), only the first day of each month is displayed.
       - **Exclusion**:
         * **Saturdays**, **Sundays** and specific days: excludes tasks completed / done on those days.
     + (**Agile Mode**) **Iterations**:
       - **Iterations**: start and end iterations as range for the X-axis of the chart. Only for tasks of both **Iteration** and **Release Backlogs** (see Type above).
   * **Dimension**: width and height of the image in pixel.
3. Check the list of tasks upon which data the chart(s) will be drawn. Tasks may be removed from the list (won’t be deleted from the database).
4. Create: the image can be saved (right click) to the local file system.

pomodoro16.png Each "bar" of the charts represents the value of the data "at the end" of the corresponding x-axis coordinate (day or iteration). Example for a burndown chart: 413 story points remaining at the end of the 5th of May. Example for burn-up chart: 51 tasks completed / done by the end of the 5th of May.

## How to create a Daily Burndown chart (Scrum)

1. Choose
   * **Burndown Chart**:
     + Type: **Story Points**
     + **Y-Legend**: eg "Story Points"
     + **X-Legend**:eg "Sprint"
     + **Target**:
       - **X-Legend**: eg "Target"
2. Configure
   * **Tasks**
     + Type: **Iteration** + Sprint number.
     + **Dates**:
       - **Dates**: start and end dates of the Sprint.
       - **Exclusion**: Saturdays, Sundays and specific days
3. Check: all tasks (done or not) must belong to the Sprint.
4. Create the Burndown chart.

## How to create a Release Product Burn-up chart (Scrum)

1. Choose
   * **Burn-up Chart**: (deselect **Burndown Chart**)
     + Type: **Story Points**
     + **Y-Legend**: eg "Story Points"
     + **X-Legend**:eg "Release product"
     + **Guide**:
       - **X-Legend**: eg "Guide"
     + **Scope**:
       - **X-Legend**: eg "Scope"
2. Configure
   * **Tasks**
     + Type: **Iteration Backlog + Release Backlog**
     + **Iterations**: start and end Sprints of the Release Product.
3. Check: all tasks must be in the range of the Release Product.
4. Create the Burn-up chart.

# Annex

## Themes

List of free Look and Feel themes used in mAP

* Java System and Cross platform themes
* Java Nimbus theme (if installed)
* NimROD
  + LGPL
  + <http://nilogonzalez.es/nimrodlf/licencia-en.html>
* JGoodies
  + BSD Open Source
  + <http://www.jgoodies.com/freeware>
* Pgs
  + Apache License 2.0
  + <http://www.pagosoft.com/projects/pgslookandfeel>
* Seaglass
  + Apache License 2.0
  + <http://code.google.com/p/seaglass>
* JTattoo
  + GPL3
  + <http://www.jtattoo.net/License.html>
* InfoNode
  + GPL
  + <http://www.infonode.se/pages/itplicense.html>

## Libraries

Main free libraries used in mAP

* SwingX
* OpenCSV
* Apache POI
* Apache POI Open XML
* JFreeChart
* Joda Time
* Google Drive API
* JDOM 2
* JSoup

## XSD schema

XSD schema for XML export and import features.

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <xs:schema version="1.0"  xmlns:xs="http://www.w3.org/2001/XMLSchema"  elementFormDefault="qualified">    <!-- group -->  <xs:group name="data">  <xs:sequence>  <xs:element name="u" type="xs:boolean"/><!-- 0 (false) or 1 (true - unplanned or interrruption) -->  <xs:element name="date" type="xs:string"/><!-- date of creation (Agile mode) or schedule date (Pomodoro mode) -->  <xs:element name="datecompleted" type="xs:string"/>  <xs:element name="title" type="xs:string"/>  <xs:element name="estimate" type="xs:nonNegativeInteger"/><!-- 0, 1, 2... -->  <xs:element name="overestimate" type="xs:nonNegativeInteger"/><!-- 0, 1, 2... -->  <xs:element name="real" type="xs:nonNegativeInteger"/><!-- 0, 1, 2... -->  <xs:element name="diffi" type="xs:integer" minOccurs="0" maxOccurs="1"/><!-- ...-1, 0, +1... : Diff I -->  <xs:element name="diffii" type="xs:integer" minOccurs="0" maxOccurs="1"/><!-- ...-1, 0, +1...: Diff II -->  <xs:element name="internal" type="xs:nonNegativeInteger"/><!-- 0, 1, 2... : number of internal interruptions -->  <xs:element name="external" type="xs:nonNegativeInteger"/><!-- 0, 1, 2... : number of external interruptions -->  <xs:element name="type" type="xs:string"/>  <xs:element name="author" type="xs:string"/>  <xs:element name="place" type="xs:string"/>  <xs:element name="description" type="xs:string"/>  <xs:element name="comment" type="xs:string"/><!-- story (Agile mode) or comment (Pomodoro mode) -->  <xs:element name="storypoints" type="xs:decimal"/><!-- 0.0, 0.5, 1.0... -->  <xs:element name="iteration" type="xs:integer"/><!-- -1, 0, 1... (Agile mode) or -1 (Pomodoro mode) -->  <xs:element name="priority" type="xs:integer"/><!-- -1, 0, 1... (Agile mode) or -1 (Pomodoro mode) -->  </xs:sequence>  </xs:group>  <!-- complex elements -->  <xs:element name="subtask">  <xs:complexType>  <xs:sequence>  <xs:group ref="data" minOccurs="1" maxOccurs="1"/>  </xs:sequence>  </xs:complexType>  </xs:element>  <xs:element name="task">  <xs:complexType>  <xs:sequence>  <xs:group ref="data" minOccurs="1" maxOccurs="1"/>  <xs:element ref="subtask" minOccurs="0" maxOccurs="unbounded"/>  </xs:sequence>  </xs:complexType>  </xs:element>  <!-- root element -->  <xs:element name="tasks">  <xs:complexType>  <xs:sequence>  <xs:element ref="task" minOccurs="1" maxOccurs="unbounded"/>  </xs:sequence>  </xs:complexType>  </xs:element>  </xs:schema> |