

 **Fix Me!** This page is not fully translated, yet. Please help completing the translation.  
(remove this paragraph once the translation is finished)

## T!M data bases view

In order to work properly with the Dashboard appropriate views have to be generated. These viewing options are presented in T!M in an accessible and easy way. The corresponding sections can be included to the MySQL Client by using copy and paste functionality (Ctrl + Enter). Another option would be to download the script and then execute this file via the MySQL Server.

[tim\\_views.sql](#)

```
CREATE DEFINER=`root`@`localhost` FUNCTION `currentClient`() RETURNS
INT(11)
  NO SQL
  DETERMINISTIC
RETURN @currentClient;
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_activity` AS SELECT
`ni`.`ID_` AS `id`,
`ni`.`NAME_` AS `name`,
`ni`.`client_ID_` AS `clientId`,
`ni`.`CREATION_TIME_` AS `creationTime`,
`ni`.`creationUser_ID_` AS `creationUserId`,
`ni`.`processInstance_ID_` AS `instanceId`,
`ni`.`realStart` AS `start`,
`ni`.`realEnd` AS `end`,
`ni`.`loopCount` AS `loopCount`,
`ni`.`duration` AS `duration`,
`ni`.`escalationTime` AS `escalationTime`,
`ni`.`estimatedEnd` AS `estimatedEnd`,
`ni`.`desiredStart` AS `milestone`,
`ni`.`desiredStartTime` AS `milestoneDate`,
`ni`.`inTime` AS `inTime`,
`ni`.`puffer` AS `timeBuffer`,
`ni`.`calFAZ` AS `calculatedEST`,
`ni`.`calFEZ` AS `calculatedEFT`,
`ni`.`calSAZ` AS `calculatedLST`,
`ni`.`calSEZ` AS `calculatedLFT`,
`ni`.`numFAZ` AS `valueEST`,
`ni`.`numFEZ` AS `valueEFT`,
`ni`.`numSAZ` AS `valueLST`,
`ni`.`numSEZ` AS `valueLFT`,
`node`.`CLASS_` AS `type`
FROM ((`LOOM_NODEINSTANCE` `ni` JOIN `LOOM_NODE` `node` ON
(`ni`.`node_ID_` = `node`.`ID_`))) WHERE (`ni`.`client_ID_` =
```

```
`currentClient`());  
  
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY  
DEFINER VIEW `view_definition` AS SELECT  
`LOOM_PROCESSEDEFINITION`.`ID` AS `id`,  
`LOOM_PROCESSEDEFINITION`.`NAME` AS `name`,  
`LOOM_PROCESSEDEFINITION`.`CLIENT` AS `clientId`,  
`LOOM_PROCESSEDEFINITION`.`CREATION_TIME` AS `creationTime`,  
`LOOM_PROCESSEDEFINITION`.`CREATION_USER` AS `creationUserId`,  
`LOOM_PROCESSEDEFINITION`.`DESCRIPTION` AS `description`,  
`LOOM_PROCESSEDEFINITION`.`VERSION` AS `version`,  
`LOOM_PROCESSEDEFINITION`.`ESCALATIONSTATUS` AS `escalationStatus`,  
`LOOM_PROCESSEDEFINITION`.`OWNER_EXPRESSION` AS `owner`,  
`LOOM_PROCESSEDEFINITION`.`STARTER_EXPRESSION` AS `starter`,  
`LOOM_PROCESSEDEFINITION`.`DEPLOYER_EXPRESSION` AS `deployer`,  
`LOOM_PROCESSEDEFINITION`.`ARCHIV` AS `archived`  
FROM `LOOM_PROCESSEDEFINITION` WHERE (`LOOM_PROCESSEDEFINITION`.`CLIENT`  
= `currentClient`());
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY  
DEFINER VIEW `view_effort` AS SELECT  
`eff`.`ID` AS `id`,  
`eff`.`NAME` AS `name`,  
`eff`.`client_ID` AS `clientId`,  
`eff`.`CREATION_TIME` AS `creationTime`,  
`eff`.`creationUser_ID` AS `creationUserId`,  
`eff`.`CLASS` AS `effortType`,  
`eff`.`effort` AS `value`,  
`eff`.`description` AS `description`,  
`eff`.`costCenter_ID` AS `costCenterId`,  
`ti`.`PROCINST` AS `instanceId`,  
`eff`.`parentFolder_ID` AS `parentFolderId`  
FROM (`LOOM_EFFORT` `eff` JOIN `LOOM_TASKINSTANCE` `ti`  
ON((`eff`.`parentFolder_ID` = `ti`.`PARENT_FOLDER`)))  
WHERE (`eff`.`client_ID` = `currentClient`());
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY  
DEFINER VIEW `view_identity` AS SELECT  
`LOOM_IDENTITY`.`ID` AS `id`,  
`LOOM_IDENTITY`.`NAME` AS `name`,  
concat(`LOOM_IDENTITY`.`namelast`, ' ', `LOOM_IDENTITY`.`namefirst`, '
```

```
(',`LOOM_IDENTITY`.`NAME`,`')') AS `displayName`,
`LOOM_IDENTITY`.`client_ID` AS `clientId`,
`LOOM_IDENTITY`.`CREATION_TIME` AS `creationTime`,
`LOOM_IDENTITY`.`creationUser_ID` AS `creationUserId`,
`LOOM_IDENTITY`.`CLASS` AS `identityType`,
`LOOM_IDENTITY`.`email` AS `email`,
`LOOM_IDENTITY`.`namefirst` AS `firstname`,
`LOOM_IDENTITY`.`namelast` AS `lastname`,
`LOOM_IDENTITY`.`ARCHIV` AS `archived`,
`LOOM_IDENTITY`.`blocked` AS `blocked`,
`LOOM_IDENTITY`.`parent_ID` AS `parentID`
FROM `LOOM_IDENTITY` WHERE (`LOOM_IDENTITY`.`client_ID` =
`currentClient`());
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_instance` AS SELECT
`pi`.`ID` AS `id`,
`pi`.`NAME` AS `name`,
`pd`.`NAME` AS `definitionName`,
`pi`.`CLIENT` AS `clientId`,
`pi`.`CREATION_TIME` AS `creationTime`,
`pi`.`key` AS `key`,
`pi`.`CREATION_USER` AS `creationUserId`,
`pi`.`PROCESSDEFINITION` AS `definitionId`,
`pi`.`DESCRIPTION` AS `instanceDescription`,
`pi`.`CREATION_GROUP` AS `creationGroup`,
`pi`.`END` AS `end`,
`pi`.`ARCHIV` AS `archiv`,
`pi`.`ARCHIVATIONUSER_ID` AS `archivationUserId`,
`pi`.`ROOTTOKEN` AS `rootToken`,
`pi`.`SUPERPROCESSTOKEN` AS `parentProcessToken`,
`pi`.`NEXT_ESCALATIONTIME` AS `nextEscalationTime`,
`pi`.`processVariableIndex_ID` AS `processVariableIndexId`,
`pi`.`inTime` AS `inTime`,
`pvi`.`field1` AS `index1`,
`pvi`.`value1` AS `value1`,
`pvi`.`field2` AS `index2`,
`pvi`.`value2` AS `value2`,
`pvi`.`field3` AS `index3`,
`pvi`.`value3` AS `value3`,
`pvi`.`field4` AS `index4`,
`pvi`.`value4` AS `value4`,
`pvi`.`field5` AS `index5`,
`pvi`.`value5` AS `value5`,
`pvi`.`field6` AS `index6`,
`pvi`.`value6` AS `value6`,
`pvi`.`field7` AS `index7`,
`pvi`.`value7` AS `value7`,
```

```
`pvi`.`field8` AS `index8`,
`pvi`.`value8` AS `value8`,
`pvi`.`field9` AS `index9`,
`pvi`.`value9` AS `value9`,
`pvi`.`field10` AS `index10`,
`pvi`.`value10` AS `value10`
FROM (((`LOOM_PROCESSINSTANCE` `pi` JOIN `LOOM_PROCESSVARIABLEINDEX`
`pvi` ON (`pi`.`processVariableIndex_ID` = `pvi`.`ID`))) JOIN
`LOOM_PROCESSEDEFINITION` `pd` ON((`pi`.`PROCESSEDEFINITION` =
`pd`.`ID`)))
WHERE (`pi`.`CLIENT` = `currentClient`());
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_swimlane` AS SELECT
`LOOM_SWIMLANEINSTANCE`.`ID` AS `id`,
`LOOM_SWIMLANEINSTANCE`.`NAME` AS `name`,
`LOOM_SWIMLANEINSTANCE`.`CLIENT` AS `clientId`,
`LOOM_SWIMLANEINSTANCE`.`PROCINST` AS `instanceId`,
`LOOM_SWIMLANEINSTANCE`.`ACTOR` AS `actor`,
`LOOM_SWIMLANEINSTANCE`.`POOLEDACTOR` AS `pooledActor`
FROM `LOOM_SWIMLANEINSTANCE` WHERE (`LOOM_SWIMLANEINSTANCE`.`CLIENT` =
`currentClient`());
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_task` AS SELECT
`ti`.`ID` AS `id`,
`ti`.`NAME` AS `name`,
`ti`.`CLIENT` AS `clientId`,
`ti`.`CREATION_TIME` AS `creationTime`,
`ti`.`CREATION_USER` AS `creationUserId`,
`ti`.`PROCINST` AS `instanceId`,
`ti`.`DESCRIPTION` AS `description`,
`ti`.`START` AS `start`,
`ti`.`END` AS `end`,
`ti`.`ISOPEN` AS `isOpen`,
`ti`.`SWIMLANEINSTANCE` AS `swimlaneId`,
`ti`.`ACTOR` AS `actor`,
`ti`.`POOLEDACTOR` AS `pooledActor`,
`ti`.`NODEINSTANCE` AS `activity`,
`ta`.`ISADHOC` AS `isAdhoc`,
`ti`.`PARENT_FOLDER` AS `parentFolderId`,
`ti`.`archiv` AS `archiv`
FROM (`LOOM_TASKINSTANCE` `ti` JOIN `LOOM_TASK` `ta` ON((`ti`.`TASK` =
```

```

`ta`.`ID_`))) WHERE (`ti`.`CLIENT_` = `currentClient`());

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_token` AS SELECT
`tok`.`ID_` AS `id`,
`tok`.`NAME_` AS `name`,
`pi`.`CLIENT_` AS `clientId`,
`tok`.`NODEINSTANCE_` AS `nodeInstance`,
`tok`.`PROCESSINSTANCE_` AS `instanceId`,
`tok`.`PARENT_` AS `parent`
FROM (`LOOM_TOKEN` `tok` JOIN `LOOM_PROCESSINSTANCE` `pi`
ON((`tok`.`PROCESSINSTANCE_` = `pi`.`ID_`))) WHERE (`pi`.`CLIENT_` =
`currentClient`());

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_variable` AS SELECT
`vi`.`ID_` AS `id`,
`vi`.`NAME_` AS `name`,
`pi`.`CLIENT_` AS `clientId`,
`vi`.`PROCESSINSTANCE_` AS `instanceId`,
`vi`.`StringValue_` AS `stringValue`,
`vi`.`label_` AS `label`
FROM (`LOOM_VARIABLEINSTANCE` `vi` JOIN `LOOM_PROCESSINSTANCE` `pi`
ON((`vi`.`PROCESSINSTANCE_` = `pi`.`ID_`))) WHERE (`pi`.`CLIENT_` =
`currentClient`());

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY
DEFINER VIEW `view_role` AS SELECT
`LOOM_IDENTITY_ID_` AS `identityId`,
`ROLE_ID_` AS `roleId`
FROM `MN_IDENTITY_ROLE`;

```

As soon these views have been created they can be used by the described [TIM table structure](#) and analyzed by [TIM Dashboard](#).

From:  
<https://wiki.tim-solutions.de/> - [TIM Wiki](#) / [NEW TIM 6 Documentation](#)

Permanent link:  
[https://wiki.tim-solutions.de/doku.php?id=en:software:dashboard:dashboard\\_views&rev=1418743919](https://wiki.tim-solutions.de/doku.php?id=en:software:dashboard:dashboard_views&rev=1418743919)

Last update: **2021/07/01 09:55**

