



# Parasoft Project Center

# PARASOFT END USER LICENSE AGREEMENT

PLEASE READ THIS END USER LICENSE AGREEMENT ("AGREEMENT") CAREFULLY BEFORE USING THE SOFTWARE. PARASOFT CORPORATION ("PARASOFT") IS WILLING TO LICENSE THE SOFTWARE TO YOU, AS AN INDIVIDUAL OR COMPANY THAT WILL BE USING THE SOFTWARE ("YOU" OR "YOUR") ONLY ON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS OF THIS AGREEMENT. THIS IS A LEGALLY ENFORCEABLE CONTRACT BETWEEN YOU AND PARASOFT. BY CLICKING THE "ACCEPT" OR "YES" BUTTON, OR OTHERWISE INDICATING ASSENT ELECTRONICALLY, OR BY INSTALLING THE SOFTWARE, YOU AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT AND ALSO AGREE THAT IT IS ENFORCEABLE LIKE ANY WRITTEN AND NEGOTIATED AGREEMENT SIGNED BY YOU. IF YOU DO NOT AGREE TO THESE TERMS AND CONDITIONS, CLICK THE "I DO NOT ACCEPT" OR "NO" BUTTON AND MAKE NO FURTHER USE OF THE SOFTWARE.

## 1.0 DEFINITIONS

1.1 "Concurrent User" means a person that has accessed the Software at any given point in time, either directly or through an application.

1.2 "Instance" means a single occurrence of initialization or execution of software on one machine.

1.3 "Licensed Capacity" means the capacity-based license pricing metrics identified in the Order Instrument, including, without limitation, Concurrent Users, Node Locked machines, and Instances.

1.4 "Maintenance" means the maintenance and technical support services for the Software identified in the Order Instrument and provided by Parasoft pursuant to this Agreement.

1.5 "Node Locked" means a license for a single machine that has been authorized to run a single Instance of the licensed Software. A Node Locked license requires that users are physically present and not accessing the machine and using the Software from a remote location.

1.6 "Software" means Parasoft's software products, in object code form, that are commercially available at the time of Your order and identified on the Order Instrument, and any modifications, corrections and updates provided by Parasoft in connection with Maintenance.

1.7 "Territory" means the country or countries in which You have a license to use the Software, as specified in Your order for the Software; or, if no Territory is specified, the country from which Your order has been issued.

1.8 "User Documentation" means the user's guide, installation guides, and/or on-line documentation applicable to the Software. User Documentation does not include marketing materials or responses to requests for proposals.

## 2.0 GRANT OF LICENSE AND USE OF SOFTWARE

2.1 License Grant. Subject to the terms and conditions of this Agreement, Parasoft grants to You a perpetual or annual (as specified in an Order Instrument), non-exclusive license to use the Software within the Territory, in accordance with the User Documentation and in compliance with the authorized Licensed Capacity. You acknowledge and agree that

this Agreement only grants a license to the Software as set forth herein and does not constitute a sale of the Software by Parasoft. You have no right to resell the Software, whether by contract or by operation of applicable copyright law.

2.2 Usage Rights. You may only use the Software and/or the User Documentation for Your internal business operations and to process Your data. You shall not (a) permit any third parties or non-licensed entities to use the Software or the User Documentation; (b) process or permit to be processed any data that is not Your data or Your affiliates' data; (c) use the Software in the operation of a service bureau; (d) sublicense, rent, or lease the Software or the User Documentation to a third party; or (e) perform, publish, or release to any third parties any benchmarks or other comparisons regarding the Software or User Documentation. You shall not make simultaneous use of the Software on multiple, partitioned, or virtual computers without first procuring an appropriate number of licenses from Parasoft. You shall not permit a third party outsourcer to use the Software to process data on Your behalf without Parasoft's prior written consent. Notwithstanding the foregoing, You may allow Your affiliates and third party contractors to use the Software solely for Your benefit; provided, however, that You shall remain fully responsible for such affiliates' and contractors' compliance with this Agreement.

2.3 License Keys. You acknowledge that the Software contains one or more license keys that will enable the functionality of the Software and third party software embedded in or distributed with the Software. You may only access and use the Software with license keys issued by Parasoft, and shall not attempt to modify, tamper with, reverse engineer, reverse compile, or disassemble any license key. If Parasoft issues a new license key for the Software, You shall not use the previous license key to enable the Software.

2.4 Archival Copies. You may make one copy of the Software for back-up and archival purposes only. You may make a reasonable number of copies of the User Documentation for Your internal use. All copies of Software and User Documentation must include all copyright and similar proprietary notices appearing on or in the originals. Copies of the Software may be stored offsite provided that all persons having access to the Software are subject to Your obligations under this Agreement and You take reasonable precautions to ensure compliance with these obligations. Parasoft reserves the right to revoke permission to reproduce copyrighted and proprietary material if Parasoft reasonably believes that You have failed to comply with its obligations hereunder.

2.5 Licensed Capacity. Parasoft licenses Software based on Licensed Capacity for different types of usage, including, without limitation, Concurrent Users and Node Locked machines. A Concurrent User license allows multiple Concurrent Users to share access to and use the Software, provided that the number of Concurrent Users accessing the Software at any time does not exceed the total number of licensed Concurrent Users. A Node Locked license allows a single specified machine to run a single Instance of the Software. If an application accessing the Software is a multiplexing, database, or web portal application that permits users of such application to access the Software or data processed by the Software, a separate Concurrent User license will be required for each Concurrent User of such application. Regardless of usage type, You shall immediately notify Parasoft in writing of any increase in use beyond the Licensed Capacity. You must obtain a license for any increase in Licensed Capacity, and You agree to pay to Parasoft additional Software license fees, which will be based on Parasoft's then-current list price.

2.6 Third Party Terms. You acknowledge that software provided by third party vendors ("Third Party Software") may be embedded in or delivered with the Software. The terms of this Agreement and any other terms that Parasoft may specify will apply to such Third Party Software, and the Third Party Software vendors will be deemed third party beneficiaries under this Agreement. You may only use the Third Party Software with the

Software. You may not use the Third Party Software on a stand-alone basis or use or integrate it with any other software or device.

2.7 Evaluation License. This Section 2.7 applies if Parasoft has provided the Software to You for evaluation purposes. Parasoft grants to You a thirty (30) day, limited license solely for the purpose of internal evaluation. You are strictly prohibited from using the Software for any production purpose or any purpose other than the sole purpose of determining whether to purchase a commercial license for the Software that You are evaluating. Parasoft is not obligated to provide maintenance or support for the evaluation Software. YOU ACKNOWLEDGE THAT SOFTWARE PROVIDED FOR EVALUATION MAY (A) HAVE LIMITED FEATURES; (B) FUNCTION FOR A LIMITED PERIOD OF TIME; OR (C) HAVE OTHER LIMITATIONS NOT CONTAINED IN A COMMERCIAL VERSION OF THE SOFTWARE. NOTWITHSTANDING ANYTHING TO THE CONTRARY IN THIS AGREEMENT, PARASOFT IS PROVIDING THE EVALUATION SOFTWARE TO YOU "AS IS", AND PARASOFT DISCLAIMS ANY AND ALL WARRANTIES (INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND STATUTORY WARRANTIES OF NON-INFRINGEMENT), LIABILITIES, AND INDEMNIFICATION OBLIGATIONS OF ANY KIND. In the event of any conflict between this Section 2.7 and any other provision of this Agreement, this Section 2.7 will prevail and supersede such other provision with respect to Software licensed to You for evaluation purposes.

2.8 Education License. If You are an educational or academic institution and are receiving a discount from Parasoft, You may use the Software solely for education or academic purposes and You may not use the Software for any commercial purpose. Parasoft may require that You provide proof of your status as an educational or academic institution.

2.9 Audit. You shall maintain accurate business records relating to its use and deployment of the Software. Parasoft shall have the right, not more than once every twelve (12) months and upon ten (10) business days prior written notice, to verify Your compliance with its obligations under this Agreement by auditing Your business records and Your use and deployment of the Software within Your information technology systems. Parasoft and/or a public accounting firm selected by Parasoft shall perform the audit during Your regular business hours and comply with Your reasonable safety and security policies and procedures. Any agreement You may require the public accounting firm to execute shall not prevent disclosure of the audit results to Parasoft. You shall reasonably cooperate and assist with such audit. You shall, upon demand, pay to Parasoft all license and Maintenance fees for any unauthorized deployments and/or excess usage of Software products disclosed by the audit. License fees for such unauthorized deployments and/or excess usage shall be invoiced to and paid by You at Parasoft's then-current list price, and applicable Maintenance fees shall be applied retroactively to the entire period of the unauthorized and/or excess usage. Parasoft shall be responsible for its own costs and expenses in conducting the audit, unless the audit indicates that You have exceeded its Licensed Capacity or otherwise exceeds its license restrictions, such that the then-current list price of non-compliant Software deployment exceeds five percent (5%) of the total then-current list price of the Software actually licensed by You, in which event You shall, upon demand, reimburse Parasoft for all reasonable costs and expenses of the audit.

3.0 TITLE. Parasoft retains all right, title and interest in and to the Software and User Documentation and all copies, improvements, enhancements, modifications and derivative works of the Software and User Documentation, including, without limitation, all patent, copyright, trade secret, trademarks and other intellectual property rights. You agrees that it shall not, and shall not authorize others to, copy (except as expressly permitted herein), make modifications to, translate, disassemble, decompile, reverse



engineer, otherwise decode or alter, or create derivative works based on the Software or User Documentation. Except as otherwise provided, Parasoft grants no express or implied rights under this license to any of Parasoft's patents, copyrights, trade secrets, trademarks, or other intellectual property rights.

#### 4.0 TERMINATION

4.1 Default; Bankruptcy. Parasoft may terminate this Agreement if (a) You fail to pay any amount when due under any order You have placed with Parasoft and do not cure such non-payment within ten (10) days of receipt of written notice of non-payment; (b) You materially breach this Agreement and do not cure such breach within thirty (30) days of receipt of written notice of such breach; (c) subject to provisions of applicable bankruptcy and insolvency laws, You become the subject of any involuntary proceeding relating to insolvency and such petition or proceeding is not dismissed within sixty (60) days of filing; or (d) You become the subject of any voluntary or involuntary petition pursuant to applicable bankruptcy or insolvency laws, or request for receivership, liquidation, or composition for the benefit of creditors and such petition, request or proceeding is not dismissed within sixty (60) days of filing.

4.2 Effect of Termination. Upon termination of this Agreement, You shall immediately discontinue use of, and uninstall and destroy all copies of, all Software. Within ten (10) days following termination, You shall certify to Parasoft in a writing signed by an officer of Yours that all Software has been uninstalled from Your computer systems and destroyed.

#### 5.0 LIMITED WARRANTY

5.1 Performance Warranty. Parasoft warrants that the Software, as delivered by Parasoft and when used in accordance with the User Documentation and the terms of this Agreement, will substantially perform in accordance with the User Documentation for a period of ninety (90) days from the date of initial delivery of the Software. If the Software does not operate as warranted and You have provided written notice of the non-conformity to Parasoft within the ninety (90) day warranty period, Parasoft shall at its option (a) repair the Software; (b) replace the Software with software of substantially the same functionality; or (c) terminate the license for the nonconforming Software and refund the applicable license fees received by Parasoft for the nonconforming Software. The foregoing warranty specifically excludes defects in or non-conformance of the Software resulting from (a) use of the Software in a manner not in accordance with the User Documentation; (b) modifications or enhancements to the Software made by or on behalf of You; (c) combining the Software with products, software, or devices not provided by Parasoft; or (d) computer hardware malfunctions, unauthorized repair, accident, or abuse.

5.2 Disclaimers. THE WARRANTIES SET FORTH IN THIS SECTION 5 ARE EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, AND PARASOFT EXPRESSLY DISCLAIMS ALL OTHER WARRANTIES, INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND STATUTORY WARRANTIES OF NON-INFRINGEMENT. PARASOFT DOES NOT WARRANT THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS OR THAT USE OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR FREE. THE REMEDIES SET FORTH IN THIS SECTION 5 ARE YOUR SOLE AND EXCLUSIVE REMEDIES AND PARASOFT'S SOLE AND EXCLUSIVE LIABILITY REGARDING FAILURE OF ANY SOFTWARE TO FUNCTION OR PERFORM AS WARRANTED IN THIS SECTION 5.

## 6.0 INDEMNIFICATION

6.1 Infringement. Parasoftware shall defend any claim against You that the Software infringes any intellectual property right of a third party, provided that the third party is located in a country that is a signatory to the Berne Convention, and shall indemnify You against any and all damages finally awarded against You by a court of final appeal, or agreed to in settlement by Parasoftware and attributable to such claim, so long as You (a) provide Parasoftware prompt written notice of the claim; (b) provide Parasoftware all reasonable assistance and information to enable Parasoftware to perform its duties under this Section 6; (c) allow Parasoftware sole control of the defense and all related settlement negotiations; and (d) have not compromised or settled such claim. If the Software is found to infringe, or if Parasoftware determines in its sole opinion that it is likely to be found to infringe, then Parasoftware may, at its option (a) obtain for You the right to continue to use the Software; (b) modify the Software to be non-infringing or replace it with a non-infringing functional equivalent, in which case You shall stop using any infringing version of the Software; or (c) terminate Your rights and Parasoftware's obligations under this Agreement with respect to such Software and refund to You the unamortized portion of the Software license fee paid for the Software based on a five year straight-line depreciation schedule commencing on the date of delivery of the Software. The foregoing indemnity will not apply to any infringement resulting from (a) use of the Software in a manner not in accordance with the User Documentation; (b) modifications or enhancements to the Software made by or on behalf of You; (c) combination, use, or operation of the Software with products not provided by Parasoftware; or (d) use of an allegedly infringing version of the Software if the alleged infringement could be avoided by the use of a different version of the Software made available to You.

6.2 Disclaimers. THIS SECTION 6 STATES YOUR SOLE AND EXCLUSIVE REMEDY AND PARASOFTWARE'S SOLE AND EXCLUSIVE LIABILITY REGARDING INFRINGEMENT OR MISAPPROPRIATION OF ANY INTELLECTUAL PROPERTY RIGHTS OF A THIRD PARTY.

7.0 LIMITATION OF LIABILITY. IN NO EVENT SHALL PARASOFTWARE OR ITS THIRD PARTY VENDORS BE LIABLE TO YOU OR ANY OTHER PARTY FOR (A) ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR (B) LOSS OF DATA, LOSS OF PROFITS, BUSINESS INTERRUPTION, OR SIMILAR DAMAGES OR LOSS, EVEN IF PARASOFTWARE AND ITS THIRD PARTY VENDORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCEPT AS LIMITED BY APPLICABLE LAW AND EXCLUDING PARASOFTWARE'S LIABILITY TO YOU UNDER SECTION 6 (INDEMNIFICATION), AND REGARDLESS OF THE BASIS FOR YOUR CLAIM, PARASOFTWARE'S MAXIMUM LIABILITY UNDER THIS AGREEMENT WILL BE LIMITED TO THE LICENSE OR MAINTENANCE FEES PAID FOR THE SOFTWARE OR MAINTENANCE GIVING RISE TO THE CLAIM. THE FOREGOING LIMITATIONS WILL APPLY NOTWITHSTANDING THE FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED REMEDY.

8.0 CONFIDENTIAL INFORMATION. For purposes of this Agreement, "Confidential Information" will include trade secrets contained within the Software and User Documentation, the terms and pricing of the Software and Maintenance (including any pricing proposals), and such other information (a) identified by either party as confidential at the time of disclosure or (b) that a reasonable person would consider confidential due to its nature and circumstances of disclosure ("Confidential Information"). Confidential Information will not include information that (a) is or becomes a part of the public domain through no act or omission of the receiving party; (b) was in the receiving party's lawful possession prior to receiving it from the disclosing party; (c) is lawfully disclosed to the receiving party by a third party without restriction on

disclosure; or (d) is independently developed by the receiving party without breaching this Agreement. Each party agrees to maintain all Confidential Information in confidence and not disclose any Confidential Information to a third party or use the Confidential Information except as permitted under this Agreement. Each party shall take all reasonable precautions necessary to ensure that the Confidential Information is not disclosed by such party or its employees, agents or authorized users to any third party. Each party agrees to immediately notify the other party of any unauthorized access to or disclosure of the Confidential Information. The receiving party agrees that any breach of this Section 8 may cause irreparable harm to the disclosing party, and such disclosing party shall be entitled to seek equitable relief in addition to all other remedies provided by this Agreement or available at law.

## 9.0 MAINTENANCE

9.1 Maintenance Period. You are required to purchase first year Maintenance with the Software, and the Maintenance period will commence upon the initial delivery of the Software and continue for a period of one year. The Maintenance period, at Your option, may be renewed pursuant to subsequent orders. Prior to such renewal, Parasoft may, upon ten (10) business days written notice, require You to provide a report on Your use and deployment of the Software. Such report will be certified by an officer of Yours and will specify, with respect to Your Software: (a) the type and amount of Licensed Capacity; (b) the version; and (c) the Parasoft license serial number. Parasoft or Parasoft's authorized distributor shall issue an annual renewal notice to You at least ninety (90) days prior to the expiration of the then-current Maintenance period. Maintenance fees will be based on the then-current list price and are subject to change without notice.

9.2 Support Coordinators. Maintenance will consist of support services provided by Parasoft or Parasoft's authorized distributor to one designated support coordinator of Yours (and one backup coordinator) per Your location, by telephone, email, or website. Support is available during normal business hours in the applicable location within the Territory, Monday through Friday, excluding nationally observed holidays.

9.3 Additional Licensed Capacity. In the event that You purchases additional Licensed Capacity for the Software prior to the annual anniversary date of the Maintenance period, You agree to pay applicable Maintenance fees based on Parasoft's then-current Maintenance rates. Maintenance fees will apply from the effective date of such additional Licensed Capacity and continue for a period of one year thereafter, unless otherwise agreed to in writing by the parties, so that Maintenance for Your previously acquired Software and added Licensed Capacity is coterminous.

9.4 New Releases. During any period in which You are current on Maintenance, Parasoft shall provide You with any new release of the Software, which may include generally available error corrections, modifications, maintenance patch releases, enhancements (unless priced separately by Parasoft and generally not included with new licenses for the Software at that time), and the revised User Documentation, if applicable. Notwithstanding the foregoing, stand-alone error corrections that are not part of a new release will not be independently supported but will be incorporated into the next release of the Software. If You install a new release of the Software, You may continue to use the previous version of the Software for up to ninety (90) days in order to assist You in the transition to the new release. Once You complete its transition to the new release of the Software, You must discontinue use of the previous version of the Software.

9.5 Supported Releases. Parasoft shall continue to support the immediately preceding release (indicated by the first number to the right of the first decimal point) of the Software for a period of twelve (12) months following the discontinuance of such Software or the date on which the new release becomes generally available, provided that You have

paid applicable Maintenance fees and incorporated all Maintenance patch releases issued by Parasoft for the release of the Software.

9.6 Reinstatement of Maintenance. If You allow Maintenance to expire, You may, at a later date, renew Maintenance by paying the following: (a) if You have installed the current release of the Software but have failed to pay the applicable renewal fee on or before the ninetieth (90th) day following expiration of the Maintenance period, annual Maintenance fees at Parasoft's then-current rates, plus Parasoft's then-current reinstatement fee; or (b) if You have not installed the current release of the Software or have failed to pay the applicable renewal fee by the ninetieth (90th) day following expiration of the Maintenance period, annual Maintenance fees at Parasoft's then-current rates, plus Parasoft's then-current license update fee for the current release of the Software.

## 10.0 GENERAL

10.1 Independent Contractors. The parties acknowledge and agree that each is an independent contractor. This Agreement will not be construed to create a partnership, joint venture or agency relationship between the parties.

10.2 Entire Agreement. The terms and conditions of this Agreement apply to all Software licensed, all User Documentation provided, and all Maintenance purchased hereunder. This Agreement will supersede any different, inconsistent or preprinted terms and conditions in any order form of Yours, purchase order or other ordering document.

10.3 Assignment. You have no right to assign, sublicense, pledge, or otherwise transfer any of Your rights in and to the Software, User Documentation or this Agreement, in whole or in part (collectively, an "Assignment"), without Parasoft's prior written consent, and any Assignment without such consent shall be null and void. Any change in control of Your organization or entity, whether by merger, share purchase, asset sale, or otherwise, will be deemed an Assignment subject to the terms of this Section 10.3.

10.4 Force Majeure. No failure, delay or default in performance of any obligation of a party to this Agreement, except payment of license fees due hereunder, will constitute an event of default or breach of the Agreement to the extent that such failure to perform, delay or default arises out of a cause, existing or future, that is beyond the reasonable control of such party, including, without limitation, action or inaction of a governmental agency, civil or military authority, fire, strike, lockout or other labor dispute, inability to obtain labor or materials on time, flood, war, riot, theft, earthquake or other natural disaster ("Force Majeure Event"). The party affected by such Force Majeure Event shall take all reasonable actions to minimize the consequences of any Force Majeure Event.

10.5 Severability. If any provision of this Agreement is held to be illegal or otherwise unenforceable by a court of competent jurisdiction, that provision will be severed and the remainder of the Agreement will remain in full force and effect.

10.6 Waiver. The waiver of any right or election of any remedy in one instance will not affect any rights or remedies in another instance. A waiver will be effective only if made in writing and signed by an authorized representative of the applicable party.

10.7 Notices. All notices required by this Agreement will be in writing, addressed to the party to be notified and deemed to have been effectively given and received (a) on the fifth business day following deposit in the mail, if sent by first class mail, postage prepaid; (b) upon receipt, if sent by registered or certified U.S. mail, postage prepaid, with return receipt requested; (c) upon transmission, if sent by facsimile and confirmation of transmission is produced by the sending machine and a copy of such facsimile is promptly sent by another means specified in this Section 10.7; or (d) upon delivery, if delivered

personally or sent by express courier service and receipt is confirmed by the recipient. Notices will be addressed to the parties based on the address stated in the applicable order, to the attention of the Legal Department. A change of address for notice purposes may be made pursuant to the procedures set forth above.

10.8 Export Restrictions. You acknowledge that the Software and certain Confidential Information (collectively "Technical Data") are subject to United States export controls under the U. S. Export Administration Act, including the Export Administration Regulations, 15 C.F.R. Parts 730 et seq. (collectively, "Export Control Laws"). Each party agrees to comply with all requirements of the Export Control Laws with respect to the Technical Data. Without limiting the foregoing, You shall not (a) export, re-export, divert or transfer any such Technical Data, or any direct product thereof, to any destination, company, or person restricted or prohibited by Export Control Laws; (b) disclose any such Technical Data to any national of any country when such disclosure is restricted or prohibited by the Export Control Laws; or (c) export or re-export the Technical Data, directly or indirectly, for nuclear, missile, or chemical/biological weaponry end uses prohibited by the Export Control Laws.

10.9 U. S. Government Rights. The Software and User Documentation are deemed to be "commercial computer software" and "commercial computer software documentation" as defined in FAR Section 12.212 and DFARS Section 227.7202, as applicable. Any use, modification, reproduction, release, performance, display, or disclosure of the Software and User Documentation by the United States government will be solely in accordance with the terms of this Agreement

10.10 Choice of Law; Jurisdiction. This Agreement is governed by and construed in accordance with the laws of the State of California, U. S. A., exclusive of any provisions of the United Nations Convention on Contracts for the International Sale of Goods, including any amendments thereto, and without regard to principles of conflicts of law. Any suits concerning this Agreement will be brought in the federal courts for the Central District of California or the state courts in Los Angeles County, California. The parties expressly agree that the Uniform Computer Information Transactions Act, as adopted or amended from time to time, shall not apply to this Agreement or the Software and Maintenance provided hereunder.

10.11 Amendment. This Agreement may only be modified by a written document signed by an authorized representative of Parasoft and by You.

10.12 Survival. Any terms of this Agreement which by their nature extend beyond the termination or expiration of this Agreement will remain in effect. Such terms will include, without limitation, all provisions herein relating to limitation of liability, title and ownership of Software, and all general provisions.

Parasoft Corporation

101 East Huntington Drive, 2nd Floor

Monrovia, CA 91016 USA

+1 (626) 256-3680

+1 (888) 305-0041 (USA only)

+1 (626) 256-9048 (Fax)

info@parasoft.com

<http://www.parasoft.com>

All brand and product names are trademarks or registered trademarks of their respective holders.

Copyright 1993-2016  
Parasoft Corporation  
101 East Huntington Drive, 2nd Floor  
Monrovia, CA 91016 USA

# PARASOFT 使用許諾契約書

本和文契約書は、英文契約書を和訳したものです。万一和英契約書の表現に不一致があった場合は英文契約書に従うこととなりますことをご了承下さい。可能な限り忠実なる対訳に努めましたが生じたご了解をいただくために必要な方は、英文契約書を必ず慎重にお読みください。

本使用許諾契約（「本契約」）をソフトウェアを使用する前に注意深くお読みください。Parasoft 社（「Parasoft」）はお客様が本契約のすべての条項に同意した場合のみ、ソフトウェアを使用する個人または企業（「お客様」あるいは「お客様の」）に対して本ソフトウェアの使用を許諾します。この契約はお客様と Parasoft との間を法的に拘束する契約です。[同意します] あるいは [はい] ボタンをクリックするかその他の方法で電子的に同意を示すこと、もしくは本ソフトウェアをインストールすることにより、お客様は本契約の諸条件に同意したものと見なされ、加えてお客様は本契約がお客様自らが書面をもって同意し署名した契約と同様の拘束力があるものであることに同意したものと見なされます。お客様がこれらの諸条件に同意されない場合は、[同意しません] あるいは [いいえ] ボタンをクリックし、本ソフトウェアをそれ以上使用しないでください。

## 1.0 用語の定義

1.1 「同時利用ユーザー」は、任意の時点で、直接あるいはアプリケーションを経由して本ソフトウェアに接続した人を意味します。

1.2 「インスタンス」は、1 台のマシン上でソフトウェアの初期化あるいは実行の単一の発生を意味します。

1.3 「ライセンス利用範囲」は、注文書に明記された料金設定による、同時利用ユーザーベース、ノードロックマシンベース、インスタンスベースを含みますがそれらに限らない最大利用数および/あるいは利用範囲を指定するライセンスを意味します。

1.4 「保守サービス」は、発注書に明記され、かつ、本契約に従って Parasoft が提供する、本ソフトウェアの保守ならびにテクニカルサポートサービスを意味します。

1.5 「ノードロック」は、許諾された本ソフトウェアの単一インスタンスを実行させる単一のマシンのためのライセンスを意味します。ノードロックライセンスはユーザーが実際に存在することを要求します。また、リモート環境でマシンに接続したり本ソフトウェアを使用したりすることはできません。

1.6 「本ソフトウェア」は、お客様が発注した時点で販売できる状態にあり、かつ、発注書類に明記されたオブジェクトコード形式の Parasoft のソフトウェア製品および保守サービスの一環として Parasoft により提供された変更、修正、アップデートを意味します。

1.7 「許諾地域」は、お客様が本ソフトウェアの発注書に本ソフトウェアを使用する権利を有する国として記載した国、もしくは、発注時に指定されていない場合はお客様の発注書が発行された国を意味します。

1.8 「ユーザードキュメント」は本ソフトウェアに関連した、ユーザーズガイド、インストールガイド、および/あるいは、オンラインドキュメントを意味します。「ユーザードキュメント」には、販促関連資料、あるいは、提案依頼に対する回答は含まれません。

## 2.0 ライセンスの許諾および本ソフトウェアの使用

### 2.1 ライセンス許諾

本契約の諸条件に従って、Parasoft はお客様に本ソフトウェアを許諾地域内で、ユーザードキュメントのとおり、許諾されたライセンス利用範囲に則って利用する永久的なあるいは年間契約（注文

書に明記されている)の非独占ライセンスを提供します。お客様は、本契約が Parasoft によってここに規定されている本ソフトウェアのライセンスを許諾する唯一のものであり、本ソフトウェアの販売を可能にするものではないことを認め、同意します。お客様は、契約によっても著作権保護法の規定によっても、本ソフトウェアを再販する権利を有することはありません。

## 2.2 使用権

お客様は、本ソフトウェアおよび/またはユーザードキュメントをお客様の内部事業運営のため、およびお客様のデータを処理するためだけに使用するがことができます。お客様は、次の (a) から (e) を行うことはできません。(a) 第三者あるいはライセンスを持たない者に本ソフトウェアあるいはユーザードキュメントを使用することを許可すること、(b) お客様あるいはお客様の関係会社のデータではないデータに対して使用すること、もしくは、使用することを許可すること、(c) 本ソフトウェアをサービスの作業で使用すること、(d) 本ソフトウェアあるいはユーザードキュメントを第三者にサブライセンス、レンタルあるいはリースすること、(e) 本ソフトウェアあるいはユーザードキュメントに係わるベンチマークあるいはその他の比較を実行する、公表する、もしくは第三者に公開すること。お客様は、Parasoft から最初に購入したライセンスの数量以外に、本ソフトウェアを複数の、または分割された、もしくは仮想化されたコンピュータで同時に使用することはできません。お客様は、Parasoft が事前に書面で同意した場合を除き、第三者のアウトソーシング業者にお客様に代わって本ソフトウェアを使用することを許可することはできません。前述にかかわらず、お客様はお客様の関係会社ならびに第三者の契約会社に対して本ソフトウェアをお客様のためにのみ使用することを許可することができます。ただし、お客様は当該関係会社および契約会社が本契約を遵守することに全責任を負います。

## 2.3 ライセンスキー

お客様は、本ソフトウェアには本ソフトウェアの機能性、ならびに本ソフトウェアとともに配布される、もしくは本ソフトウェアに組み込まれているサードパーティソフトウェアの機能性を有効にする1つあるいはそれ以上のライセンスキーが含まれることを了承します。お客様が Parasoft より発行されたライセンスキーでのみ、本ソフトウェアに接続し使用することができます。お客様は、ライセンスキーを編集、改ざん、リバースエンジニアリング、逆コンパイル、もしくは逆アセンブリすることはできません。Parasoft から本ソフトウェアの新しいライセンスキーが発行された場合は、お客様は以前に発行された本ソフトウェアの実行を可能にするライセンスキーを使用することはできません。

## 2.4 保存用複製

お客様は、本ソフトウェアのコピーを1つ、バックアップあるいは保存の目的でのみ作成することができます。お客様は、ユーザードキュメントについて妥当な数のコピーをお客様の内部使用のため作成することができます。本ソフトウェアおよびユーザードキュメントのすべての複製には、原本に表示されている著作権および類似する財産権によって保護されている旨の通知を含めなければなりません。本ソフトウェアに係わるすべての人が本契約で規定されているお客様の義務を負い、かつ、お客様がこれらの義務を遵守するため合理的な予防措置を講ずるならば、本ソフトウェアの複製を離れた場所に保管することができます。Parasoft は、Parasoft がお客様が本契約で規定されている義務の遵守を怠ったと信じる合理的な根拠が存在する場合、著作権ならびに財産権で保護されている製作物の複製を作成する許可を取り消す権利を有します。

## 2.5 ライセンス利用範囲

Parasoft は、本ソフトウェアのライセンスをライセンス利用範囲に基づいて同時利用ユーザーあるいはノードロックマシンを含むがこれらに限らないさまざまな利用形態で提供します。同時利用ユーザーライセンスは、本ソフトウェアに接続する同時利用ユーザー数が購入した同時利用ユーザー数の合計を上回らない限り、複数の同時利用ユーザーが本ソフトウェアの接続を共有し使用することができます。ノードロックライセンスは、単独の特定のマシンで本ソフトウェアの単一インスタンスを実行することができます。本ソフトウェアが接続するアプリケーションがそのアプリケーションのユーザーに本ソフトウェア、もしくは本ソフトウェアによって処理されたデータへの接続を許可する、多重処理方式、データベースもしくは Web ポータルアプリケーションであった場合、各アプリケー



ションに対して別個の同時利用ユーザーライセンスが要求されます。利用形態に係わらず、お客様は、速やかに Parasoft にライセンス利用範囲を超えた利用の増加を書面で通知するものとします。お客様は、ライセンス利用範囲に係わるライセンスの増加分のライセンスを取得しなければなりません。また、お客様は、本ソフトウェアの追加ライセンスに対する費用を Parasoft のその時点の最新の価格表に則って Parasoft に支払うことに同意するものとします。

## 2.6 第三者の条件

お客様は、サードパーティベンダー（「サードパーティのソフトウェア」）によって提供されるソフトウェアが本ソフトウェアとともに提供されること、もしくは本ソフトウェアに組み込まれていることを了承します。本契約の諸条件と Parasoft が特定するその他の条件は、当該サードパーティのソフトウェアに適用され、サードパーティのソフトウェアのベンダーは本契約に基づいて第三受益者と見なされます。お客様は、サードパーティのソフトウェアを本ソフトウェアとともにのみ使用することができます。お客様は、サードパーティのソフトウェアを単独で使用すること、またはその他のソフトウェアもしくはデバイスと結合あるいは使用することはできません。

## 2.7 評価用ライセンス

この 2.7 章は、Parasoft が本ソフトウェアをお客様に評価目的で使用するものとして提供した場合に適用されます。Parasoft は、お客様にお客様の組織内での評価を目的とするためだけに 30 日間の制限付きライセンスを提供します。お客様は、本ソフトウェアを製造目的で使用すること、もしくはお客様が評価している本ソフトウェアの商用ライセンスを購入するかどうかの判断のため以外の目的で使用することを厳密に禁止されています。Parasoft は、本ソフトウェアの評価使用に対する保守サービスあるいはサポートを提供する義務はありません。お客様は、評価用の本ソフトウェアは、(A) 機能に制限があること、(B) 時間的な制限があること、(C) 本ソフトウェアの商用バージョンには含まれないその他の制限があることを了承します。本契約の他の規定に係わらず、Parasoft は、評価用の本ソフトウェアをお客様に、「現状のまま」で提供し、Parasoft は、いかなる保証（商品性、特定目的への適合性、第三者の権利を侵害していないことに対する保証を含みますがこれに限りません。）、責任、その他のあらゆる賠償義務を負いません。この 2.7 章と本契約の他の条項との間に不一致がある場合は、評価目的のためお客様に提供された本ソフトウェアについては、この 2.7 章が優先され、かかるその他の条項は本条項に置き換えられます。

## 2.8 教育機関向けライセンス

お客様が教育あるいは学術研究機関である場合、Parasoft よりディスカウントを受けることができます。お客様は本ソフトウェアを教育あるいは学術研究のみを目的として使用することができますが、お客様は本ソフトウェアを営利を目的として使用することはできません。Parasoft は、お客様が教育あるいは学術研究機関であることの証明を要求することができます。

## 2.9 監査

お客様は、本ソフトウェアの使用および配布に係わる取引記録を正確に保管するものとします。Parasoft は、12 か月につき 1 回以下の割合で 10 営業日前の書面による通知をもって、お客様の本ソフトウェアの取引記録ならびに情報処理システム内の本ソフトウェアの使用および配布について監査することにより、お客様の本契約における義務の遵守を検証する権利を有するものとします。Parasoft および / または Parasoft によって指定された公認会計士事務所は、お客様が通常の営業時間に監査を実施するものとし、お客様の適切な安全および保安ポリシーならびに手順を遵守するものとし、お客様が公認会計士事務所を実施を要求できるいかなる契約があつたとしても、Parasoft への監査結果の報告を拒否することは認められません。お客様は、当該監査に協力・支援するものとします。お客様は、必要に応じて、Parasoft に監査によって明らかになった許諾されていない配布および / または超過した使用に利用された本ソフトウェア製品についてすべてのライセンスと保守サービスの費用を支払うものとします。許諾されていない配布および / または超過した使用に該当するライセンスの費用は請求され、お客様により Parasoft のその時点の最新の価格で支払われるものとします。また、適用される保守サービス費用は、許諾されていない配布および / または超過した使用の全期間について、さかのぼって適用されます。Parasoft は、監査により、お客様がライセンス

利用範囲を超えて利用していたことが確認された、もしくはライセンスの制限数を超過していたことが明らかになった場合を除き、監査の実施に伴う費用と支出を負担します。お客様が許諾されていない本ソフトウェアを配布したり、超過して使用していた場合、許諾されていない本ソフトウェアの配布についてその時点の最新の価格表に則って算出した金額が、実際にお客様に許諾されているすべての本ソフトウェアについてその時点の最新の価格表に則って計算した合計金額の5%を超えていた場合、お客様は、必要に応じて、監査にかかった費用と支出の全額をParasoftに返済するものとします。

### 3.0 権原

Parasoftは、本ソフトウェアおよびユーザードキュメントについて、複製、改善、機能追加、変更、本ソフトウェアおよびユーザードキュメントの派生物に係わる、すべての特許、著作権、企業秘密、登録商標、その他の知的財産権を含むがこれらに限らないすべての権利、権原、利益を有します。お客様は、本ソフトウェアあるいはユーザードキュメントについて、複製(本契約に明記されるものを除く)、変更の作成、翻訳、逆アセンブリ、逆コンパイル、リバースエンジニアリング、デコードあるいは改変、派生物の作成を実施することは認められていないことに同意します。また、お客様はこれらについて他者に実施の許可を与えることも認められていないことに同意します。特別の定めがある場合を除き、Parasoftは、明示あるいは暗黙を問わず、Parasoftの特許、著作権、企業秘密、登録商標、その他知的財産権のいずれかに係わる権利を許諾しません。

### 4.0 契約の解除

#### 4.1 不履行; 破産

Parasoftは、次の(a)から(d)のいずれかの状態になったとき、本契約を解除することができます。(a)お客様がParasoftに発行した注文書に記載されている支払期日に全額が支払われず、書面による未払いの通知を受け取ってから10日以内に未払いが是正されない、(b)お客様が本契約に対して重大な違反をし、書面による違反の通知を受け取ってから30日以内にかかる違反が是正されない、(c)適用可能な破産法あるいは倒産法の条項に基づき、お客様が倒産に伴う強制的手続きの対象になり、該当する申し立てあるいは手続きが60日以内に終了しない、(d)お客様が適用可能な破産法あるいは倒産法に基づき自発的あるいは強制的な申し立ての対象になる、もしくは債権者のため倒産処理手続、清算手続、債務減免手続を要求する場合で、かつ、該当する申し立て、要求、手続きが、60日以内に終了しない。

#### 4.2 解除の履行

本契約を解除された場合には、お客様は速やかに本ソフトウェアの使用を中止し、アンインストールし、すべての複製を破棄するものとします。解除から10日以内にお客様はParasoftにすべての本ソフトウェアがお客様のコンピュータシステムからアンインストールされ破棄されたことを証明する書類にお客様の役員が署名したものを送付するものとします。

### 5.0 限定保証

#### 5.1 動作保証

Parasoftは、本ソフトウェアがParasoftから納品され、かつ、ユーザードキュメントと本契約の諸条件に従って使用された場合に、本ソフトウェアがユーザードキュメントの記載内容のとおり実質的に動作することを、本ソフトウェアの最初の納品日から90日間保証します。本ソフトウェアが、保証内容のように動作せず、お客様が90日間の保証期間の間にParasoftに不具合を书面で通知した場合は、Parasoftは、任意の選択として、次の(a)から(c)を実施するものとします。(a)本ソフトウェアを修正する、(b)本ソフトウェアを実質的に同じ機能を持つソフトウェアに交換する、(c)不具合のある本ソフトウェアのライセンスを終了し、Parasoftが不具合のある本ソフトウェアの対

価として受け取ったライセンスの費用を払い戻す。前述の保証は、明確に本ソフトウェアの欠陥あるいは不具合が次の (a) から (d) によるものの場合、保証の対象から外します。(a) ユーザードキュメントに反する本ソフトウェアの使用、(b) お客様による本ソフトウェアへの変更あるいは機能追加、(c) 本ソフトウェアと Parasoft によって提供されたものでない製品、ソフトウェアもしくはデバイスとの統合、(d) コンピュータハードウェアの故障、認められていない修理、事故、誤用のいずれか。

## 5.2 免責事項

5章に明記されている保証は、唯一の、すべての他の保証に代わるものであり、Parasoft は、明示あるいは暗黙を問わず、すべてのその他の一切の保証の責任を負いません。これには、商品性、特定目的への適合性、第三者の法的権利を侵害していないことを含みますがこれに限りません。Parasoft は、本ソフトウェアがお客様の要求を満たす、もしくは本ソフトウェアを途切れることなく使用できる、あるいはエラーのない状態になることを保証しません。5章に明記されている救済措置は、5章に関するお客様に対する唯一の救済措置であり、5章で保証するすべてのソフトウェアの実行あるいは実施の失敗に係わる Parasoft の唯一の保証です。

## 6.0 補償

### 6.1 侵害

Parasoft は、第三者がベルヌ条約に加盟している国に所在するならば、本ソフトウェアが第三者の知的財産権を侵害しているというお客様に対する請求について防御するものとし、上告裁判所によって最終的にお客様に裁定が下った場合の損害、Parasoft によって和解が成立した当該請求に起因する損害について、お客様が次の (a) から (d) を守る限り、賠償するものとします。(a) Parasoft に速やかに書面で請求の通知を送付する、(b) Parasoft に Parasoft が 6 章に基づく義務を果たすために必要な合理的な支援と情報を提供する、(c) Parasoft に独自の弁護活動とすべての和解交渉活動の統括を許可する、(d) 和解または示談をしない。本ソフトウェアが侵害していることを発見された、もしくは、Parasoft が侵害している可能性があるとして独自に判断したならば、Parasoft は、任意の選択として、次の (a) から (c) の措置を講ずることができます。(a) お客様に本ソフトウェアの使用を継続する権利を提供する、(b) お客様が侵害しているバージョンの本ソフトウェアの使用を中止できるよう、本ソフトウェアを侵害していない状態に修正する、もしくは機能的に同等なもので侵害していないものに置き換える、(c) 該当する本ソフトウェアについて本契約で規定されているお客様の権利と Parasoft の義務を終了し、お客様に本ソフトウェアのライセンス購入額の未償却分として 5 年の減価償却定額法をもとに本ソフトウェアの納品日から起算した額を払い戻す。前述の賠償は、次の (a) から (d) の結果として発生した侵害には適用されません。(a) ユーザードキュメントに反する本ソフトウェアの使用、(b) 本ソフトウェアに対してお客様によって、あるいは、お客様のために施された変更あるいは機能追加、(c) Parasoft から提供されたものでない製品と本ソフトウェアの統合、使用、操作、(d) 侵害の申し立てが、本ソフトウェアのお客様が使用できる他のバージョンを使用することにより回避できるにも係わらず、侵害しているバージョンの本ソフトウェアを使用すること。

### 6.2 免責事項

6章は、第三者の知的財産権に対する侵害または不正目的使用に係わるお客様に対する唯一の救済措置および Parasoft の唯一の保証です。

## 7.0 責任制限

Parasoft あるいはそのサードパーティベンダーは、お客様あるいはその他の当事者に、(A) いかなる特別、付随的、間接的、または結果的損失、(B) データの喪失、利益の損失、事業の中断、類似の経済的損失あるいは損害に対する一切の責任を負いません。これは、Parasoft あるいはサードパーティベンダーがかかる損害が発生する可能性を予め警告されていたとしても変わりません。適用法と

6章(補償)に基づく Parasoft のお客様への唯一の賠償責任を除いて、また、お客様の請求の基礎に関係なく、本契約に基づく Parasoft の最大賠償額は、請求の契機となった本ソフトウェアあるいは保守サービスの対価としてお客様が支払った本ソフトウェアあるいは保守サービスの額を超えることはありません。前述の制限は、救済措置の主たる目的を果たすことができるかどうかに係わらず、適用されます。

## 8.0 機密情報

本契約の解釈上、機密情報には、本ソフトウェアおよびユーザードキュメントに含まれる企業秘密、本ソフトウェアおよび保守サービスの諸条件あるいは価格(提案価格を含む)、ならびに、次の(a)および(b)のようなその他の情報、(a)いずれかの当事者によって開示の時点で機密であると判断された情報、(b)一般的にその性質や開示の状況から機密であると考えられる情報、が含まれます(「機密情報」)。機密情報には、次の(a)から(d)に該当する情報は含まれません。(a)公知の情報、もしくは受領者の作為あるいは不作為による過失によって公知となった情報、(b)受領者が開示者から得る以前から法的に正当に所有していた情報、(c)開示の制限なしに第三者によって受領者に法的に正当に開示された情報、(d)受領者によって本契約に違反することなく独自に開発された情報。両当事者は、本契約において認められている場合を除いて、すべての機密情報を機密に管理すること、ならびに機密情報を第三者に開示しないこと、あるいは機密情報を使用しないことを同意します。両当事者は、機密情報がかかる当事者あるいは従業員、もしくは、代理人あるいは正規ユーザーによって第三者に漏洩されないことを保証するのに必要なすべての合理的な予防措置を施すものとします。両当事者は、相手方に機密情報への不正なアクセス、あるいは、機密情報の漏洩の発生を速やかに通知することに同意します。受領者は、本8章に対する違反は開示者に回復することができない損害を与える可能性があることを了承します。かかる開示者は本契約あるいは適用法で定められたすべてのその他の救済措置に加えて法的救済を求める権利を有するものとします。

## 9.0 保守サービス

### 9.1 保守サービスの期間

お客様は、本ソフトウェアの購入と同時に最初の1年間の保守サービスを購入する必要があります。保守サービスの期間は、本ソフトウェアの最初の納品日から1年間です。保守サービス期間は、お客様の任意で更新することができます。Parasoftは、かかる更新の10営業日前の書面による通知で、お客様に本ソフトウェアに関するお客様の使用および配布に関するレポートの提出を要求することができます。かかるレポートは、お客様の役員によって承認されたものであり、かつ、お客様の本ソフトウェアについて、次の(a)から(c)に規定されている内容を明記するものとします。(a)利用形態およびライセンス利用範囲の合計、(b)バージョン、(c)Parasoft ライセンス シリアル ナンバー。Parasoft あるいは Parasoft の正規販売代理店は、お客様に年間保守サービスの更新に係わる通知を遅くとも保守サービスの終了日の90日前までに発行するものとします。保守サービス費用は、その時点での最新の価格表に則って算出されます。また、保守サービス費用は、通知なしに変更される場合があります。

### 9.2 サポート担当者

保守サービスは、Parasoft あるいは Parasoft の正規販売代理店によるサポートサービスで、お客様の所在地につき、1名の指定サポート担当者(加えて1名の副サポート担当者)に対して、電話、電子メール、Webサイトのいずれかで提供されます。サポート対応時間は、ライセンス利用範囲で指定された地域の祝日を除く、月曜日から金曜日、通常の営業時間です。

### 9.3 追加のライセンス利用範囲

お客様が以前に導入した本ソフトウェアのライセンス利用範囲の年間保守サービスが終了日を迎える前にライセンス利用範囲を追加購入する場合、お客様は追加購入分の保守サービスの費用について Parasoft のその時点での最新の保守サービス率に則って支払うことに同意します。保守サービス費

用は、お客様が先に購入した本ソフトウェアの保守サービスと追加購入のライセンス利用範囲の保守サービスの終了日を一致させるよう両当事者が書面で同意しない限り、追加購入のライセンス利用範囲の有効日から1年間が適用されます。

#### 9.4 新リリース

お客様が保守サービスに加入している間、Parasoftは、その時点で可能であれば、一般的に対応可能なエラーの修正、変更、パッチ、機能追加 (Parasoftによって別料金が課せられるもの、あるいは、その時点で本ソフトウェアに正式に含まれないものを除く)、ユーザードキュメントの改訂版を含む本ソフトウェアのあらゆる新リリース版を提供するものとします。前述にかかわらず、新リリース版に含まれないエラーの修正は、個別にサポートされることはありませんが、本ソフトウェアの次のリリース版に組み込まれる可能性があります。お客様は、本ソフトウェアの新リリース版をインストールした場合、新リリース版への移行をスムーズに行えるよう本ソフトウェアの前バージョンを90日間引き続き使用することができます。お客様が本ソフトウェアの新リリース版への移行を完了した場合には、お客様は本ソフトウェアの前バージョンの使用を終了しなければなりません。

#### 9.5 サポート対象リリース

Parasoftは、お客様が該当する保守サービス費用を支払っていて、かつ、Parasoftによって提供されたすべてのパッチを本ソフトウェアに適用しているならば、本ソフトウェアの直前のリリース版 (そのバージョンはバージョン番号の最初の小数点の右にある数字によって示される) のサポートを、本ソフトウェアの廃止から12か月間、もしくは、新リリース版が正式なものとなった日から12か月間、継続するものとします。

#### 9.6 保守サービスの再加入

お客様が保守サービスを更新しなかった場合、お客様は、後日、次の(a)と(b)のいずれかを支払うことによって、保守サービスに再加入することができます。(a)お客様は本ソフトウェアの最新リリース版をインストールしているが、該当する更新費用を保守サービスの終了日から90日以内に支払わなかった場合は、Parasoftのその時点での最新の保守サービス率に則った年間保守サービス費用、加えて、Parasoftのその時点での最新の再加入費用、(b)お客様が本ソフトウェアの最新リリース版をインストールしていない場合、もしくは、該当する更新費用を保守サービス期間が終了してから90日を過ぎても支払わなかった場合、Parasoftのその時点での最新の保守サービス率に則った年間保守サービス費用、加えて、本ソフトウェアを最新リリース版にするためのParasoftのその時点での最新のアップデート費用。

### 10.0 一般条項

#### 10.1 当事者間の独立

両当事者は、互いに独立の契約者であることを認め、同意します。本契約は、両当事者の間にパートナーシップあるいは合弁事業もしくは代理関係を創設すると解釈されることはありません。

#### 10.2 完全合意

本契約の諸条件は、本契約に基づいて購入した、すべての本ソフトウェアのライセンス、すべてのユーザードキュメント、すべての保守サービスに適用されます。お客様の注文書、発注書、もしくはその他注文に係わる書類に本契約と一致しないあるいは矛盾した、もしくは予め印字された諸条件については、本契約が優先され、取って代わるものとします。

#### 10.3 譲渡

お客様には、Parasoftが事前の書面で同意した場合を除き、本ソフトウェア、ユーザードキュメント、本契約に対する、あるいは、本契約に含まれるお客様の権利をすべてあるいは一部でも譲渡する、サブライセンスする、質権にする、移転する権利はありません(「譲渡」)。かかる同意を得ていない譲渡は、無効であるものとします。合併、買収、売却、あるいは他の原因により、お客様の組織あるいは企業の支配権が移転した場合でも、10.3章にある譲渡と見なされます。

#### 10.4 不可抗力

いずれかの当事者が本契約に基づくライセンス費用の支払いを除く本契約による義務の履行の失敗、遅延、不履行を犯したとしても、本契約による義務の不履行あるいは違反が存在するあるいは将来生ずるかかかる当事者の合理的な制御を超える事象に起因する、履行の失敗、遅延、不履行であるならば、かかる当事者に責任はないものとします。その事象とは、政府機関、市民・軍事機関による作為あるいは不作為、火災、ストライキ、ロックアウト、その他の労働争議、労働者あるいは材料の受け入れ困難、洪水、戦争、暴動、窃盗、地震、自然災害（「不可抗力」）を含みますがこれらに限りません。不可抗力の影響を受けた当事者は不可抗力の影響を最小限に留めるため、すべての合理的な措置を施すものとします。

#### 10.5 可分性

本契約の条項が法律に反している、もしくは、管轄裁判所によって強制不能であると判断された場合でも、その条項を切り離れた本契約の残りの条項は完全効力が付与されるものとします。

#### 10.6 権利の放棄

1つの事案についての権利あるいは救済措置の公示の放棄は、他の事案の権利あるいは救済措置には影響しません。放棄は書面に記載され、当該当事者の役員によって署名されたもののみが有効と見なされます。

#### 10.7 通知

本契約に基づくすべての通知は書面でなされるものとし、次の (a) から (d) のいずれかをもって、通知される当事者の授受が有効だと思われる送付先に送付されたものと見なされます。(a) 第1種郵便かつ郵便料金前払いで投函された場合は投函から5営業日、(b) アメリカ合衆国郵便公社の書留郵便あるいは書留配達便で、かつ、郵便料金前払いの配達証明付きで配達された場合は受け取り、(c) ファクシミリで送信され、送信確認装置が搭載されているファクシミリ機から送信された、もしくは、かかるファクシミリのコピーが10.7章に規定された他の方法で即座に送付された場合は送信、(d) 人により配達され、もしくは、速配サービスで送付され、受け取りが受取人によって確認された場合は受け取り。いずれの当事者の正式な注文に記載されている住所あるいは法務部門あてに通知が送付されるものとします。通知先住所の変更は上記の手続きにより実施することができます。

#### 10.8 輸出制限

お客様は、本ソフトウェアと一定の機密情報（「技術データ」）は輸出管理規則（15 CFR Part 730 et. seq.）を含む米国輸出管理法（「輸出管理法」）に基づく米国輸出規制の対象であることを認識します。両当事者は技術データについて輸出管理法のすべての規制に従うことに同意します。前述に限定することなく、お客様は次の (a) から (c) を実施することはできません。(a) 技術データあるいはその直近の製品を輸出管理法で禁止あるいは規制されている地域、企業、個人に対して輸出、再輸出、移転、譲渡すること、(b) 輸出管理法でかかる開示が禁止あるいは規制されている場合、技術データをあらゆる国のあらゆる国民に対して開示すること (c) 技術データを、間接的か直接的かを問わず、輸出管理法で禁止されている、核兵器、ミサイル、化学兵器 / 生物兵器などの武器の最終用途のために輸出あるいは再輸出すること。

#### 10.9 米国政府による規制

本ソフトウェアとユーザードキュメントは、該当するならば、アメリカ合衆国政府による FAR Section 12.212 および DFARS Section 227.7202 に定義されている「商用コンピューターソフトウェア」および「商用コンピューターソフトウェアドキュメント」と見なされます。米国政府による本ソフトウェアとユーザードキュメントのいかなる使用、変更、再生、発表、実演、展示、公開は、本契約の諸条件に従ってのみ、行われるものとします。

#### 10.10 準拠法；管轄権

本契約は、覚書を含む国際物品売買契約に係わる国際連合条約のあらゆる条項を除いて、米国カリフォルニア州法に準拠し、抵触法の原則は適用されません。本契約に係わる控訴は、カリフォルニア中央地区連邦裁判所あるいはカリフォルニアロサンジェルス地区州裁判所によって行われるものとし

ます。両当事者は、統一コンピュータ情報取引法 (UCITA) が適宜、修正され、承認されるままに、ここに規定されている本ソフトウェアおよび保守サービスあるいは本契約に適用されることはないものとするに明白に同意します。

#### 10.11 覚書

本契約は、書面に記載され、Parasoft およびお客様の役員によって署名されたもののみ、修正することができます。

#### 10.12 存続

本契約の終了あるいは解除後も性質上存続すべき本契約の条項は有効に存続するものとします。かかる条項には、制限なく、責任制限、本ソフトウェアの権原および所有権に係わるすべての条項ならびにすべての一般条項が含まれます。

Parasoft Corporation

101 East Huntington Drive, 2nd Floor

Monrovia, CA 91016 USA

+1 (626) 256-3680

+1 (888) 305-0041 (USA only)

+1 (626) 256-9048 (Fax)

info@parasoft.com

<http://www.parasoft.com>

すべての商標ならびに製品名は、それぞれの商標権者の商標マークまたは登録商標マークです。

Copyright 1993-2016

Parasoft Corporation

101 East Huntington Drive, 2nd Floor

Monrovia, CA 91016 USA

---

# 目次

Project Center とは .....	4
<b>DTP のインストール</b>	
パスワードの取得 .....	7
Project Center の用語 .....	8
<b>チーム メンバーにとってのメリット</b>	
マネージャーによる Project Center の利用 .....	10
開発者による Project Center の利用 .....	12
QA チームによる Project Center の利用 .....	13
<b>一般的なユース ケースとワークフロー</b>	
Project Center でのプロジェクト管理：一般的なケース .....	15
プロジェクト計画 .....	17
進行中のプロジェクト .....	27
開発プロセスの改善 .....	38
<b>Project Center の基本操作</b>	
GUI の詳細 .....	41
Project Center での検索操作 .....	51
CSV へのデータのエクスポート .....	54
タスクのタグ付け .....	56
自動テストと Project Center との関連付け .....	60
Project Center 成果物 (欠陥、要求、タスク、テスト) の関連付け .....	70
ストーリー ポイント .....	73
電子署名 / 変更理由の入力とトラッキング .....	75
<b>Project Center の要求と欠陥 / 改善</b>	
要求の作成と管理 .....	79
欠陥 / 改善 .....	93
要求と欠陥 / 改善のタスク化 .....	106
<b>Project Center のタスクとイテレーション</b>	
イテレーションへのプロジェクトの分割 .....	112
イテレーションの計画 .....	114
タスクの新規作成 .....	123
タスクの管理 .....	125
CSV からのタスクのインポート .....	130
予測作業時間の変更 .....	134
タスクの検索 .....	135
イテレーションの編集 .....	137
イテレーション内のタスクの管理 .....	141
<b>Project Center でのテスト</b>	
手動テスト シナリオの追加と編集 .....	147
テスト シナリオのインポート .....	154
受け入れテスト セッションの実行 .....	158
Project Center からの SOAtest の実行 .....	168
<b>Project Center のレポート</b>	
プロジェクト ステータス レポート .....	173



要求レポート .....	176
欠陥 / 改善レポート .....	186
タスク レポート .....	191
イテレーションレポート .....	197
テスト レポート .....	201

## 付録 : Project Center

Microsoft Project の統合 .....	207
MS Project アドインのインストール : トラブル シューティングとヒント .....	218
タスク、欠陥、要求の電子メール通知 .....	223
バグ トラッキング システムとの同期 .....	226
作業中のタスクの経過時間 ( 実時間 ) の計算方法 .....	227

## 管理者

プロジェクトの作成と構成 .....	230
ワークフロー、権限、承認プロセスの構成 .....	241
Report Center と Project Center のコンポーネント .....	243
データベース接続の変更 .....	245
[User Administration] ページ .....	249
Project Center 管理ページ .....	272
Data Collector の構成 .....	286
Report Center と Project Center のコンポーネントのポート変更 .....	287
Cache Report Executor の設定 .....	289

## Team Server の設定

テクニカル サポート アーカイブの作成 .....	294
テクニカル サポート .....	295

## License Server の使用

### License Server の設定

License Server の設定 .....	298
--------------------------	-----

### ライセンスの管理

ライセンスの追加と管理 .....	302
ライセンス サマリ .....	312
サポートされているツールの表示 .....	316
アクセス ログの表示 .....	318

### その他の構成

制限モードでの権限とステータス遷移の設定 .....	321
Microsoft Word へのエクスポートのカスタマイズ .....	325
信頼された証明書の追加 .....	326
ツールバーでのアプリケーションの無効化と有効化 .....	327

## Development Testing Platform の統合

Parasoft Test から Report Center/Project Center へのテスト結果の送信 .....	329
CSV からの BTS データのインポート .....	330
バグ トラッキング システムおよび要求管理システムの統合 .....	336
BTS/RMS Scanner と Updater .....	342

## Report Center および Project Center とサードパーティ製ツールの統合

HP Quality Center との統合 .....	357
Bugzilla との統合 .....	365

IBM Rational ClearQuest との統合 .....	368
Atlassian JIRA との統合 .....	373
IBM Rational Change および Rational Synergy との統合 .....	381
コード レビューの統合 .....	386
Report Center と Emma の統合 .....	389
Report Center とソース管理システムの統合 .....	392
PTC Integrity ソース管理コントロール拡張との統合 .....	396
サードパーティのツールから DTP へのテスト結果の送信 .....	405
ライセンスの変更 .....	406
DNS サービスの設定 .....	407
MySQL サーバーのアップグレード .....	409
Configuration Manager の内部詳細 (Linux/Solaris) .....	410
Linux での初期データベースの作成とデータベースのアップグレード ( コマンドライン メニューからの 実行 ) .....	412
crontab による自動起動の設定 (Linux) .....	414
<b>トラブルシューティング</b>	
ログ ファイルの参照 .....	416
Cache Report Executor .....	418
索引 .....	i

---

# Project Center とは

Project Center は、開発チームの作業品質と作業効率を向上する、高度なタスク管理を可能にします。

- マネージャーがプロジェクトを計画したり、プロジェクトの進捗をトラッキングしたり、ソフトウェア開発ライフ サイクル (SDLC) プロセスを改善するために完了後のプロジェクトを解析したりするのを補助します。
- 正確な計画を開発者に提供します。開発者は、全体的なプロジェクト目標、目標を達成するために完了しなければならない自分のタスク、および各タスクの期限を把握できます。
- 開発の進捗とプロジェクト成果物 ( 要求、タスク、欠陥 / 改善 ) の実装ステータスを QA チームに提供します。QA チームはテストの準備ができていない物を把握できます。

ライセンスによっては、Project Center にアクセスできない場合があります ( 詳細については 311 ページの「全機能の DTP ライセンスへのアップグレード」を参照してください )。Project Center の使用に興味がある場合は、Parasoft 担当者に問い合わせてください。

---

## 集約された要求と課題トラッキング

Parasoft のソリューションの目的は、プロジェクトの要求、欠陥 / 改善、およびタスクを 1 つのシステムにまとめることによって、プロジェクトの計画、プロセスのモニタリング、およびチームの参画を単純化して強化することです。ほとんどのバグ トラッキング システム (BTS)、要求管理システム (RMS)、および Microsoft Project と統合し、機能性を拡張してコンポーネント同士を関連付けます。さらに、Development Testing Platform には独自の要求リポジトリがあり、これも製品バックログの役割を果たします。この要求リポジトリは、バックログ アイテムの優先順位付けを実現し、要求の階層化をサポートします。

---

## リリース計画

Project Center は、開発チームが期待される機能とスケジュールに対して妥当なリリース目標を計画するのを補助します。実行できて計測できる作業タスクに要求と課題が分割されます。個々の作業タスクについて、タスクの完了にかかる費用と時間がチーム メンバーによって見積られます。

必ず守らなければならない期限がある場合、Project Center は、現在のリソースを前提として、期限までにどの機能を無理なく完成できるかを開発チームが判断するのを補助します。期限がもっと柔軟な場合、Project Center は機能を完成するために必要な時間をその場で見積ります。その結果を見て、開発チームは関係者やマネージャーと話し合い、現実的な選択肢を決定することができます。

---

## イテレーションの計画

次に、タスクをイテレーションまたはスプリントとしてスケジューリングすることができます。開発チームによる見積りを利用して、Development Testing Platform はタスクのさまざまな組み合わせについて必要な作業時間 ( および予測される期限 ) をその場で計算します。これは、開発チームがリリースをイテレーションに分割するのに役立ちます。

---

## リリース準備

リリースを出す前に、開発チームはそのリリースで計画されたユーザー ストーリーが正しく実装されたかどうかを評価する必要があります。Development Testing Platform はマルチ レベルでこれを補助します。まず、関連する開発タスクとテスト タスクが完了としてマークされているかどうかをモニタリングします。さらに、特定の要求を検証するテストの結果とカバレッジを表示します。そして、信頼性、セキュリティ、パフォーマンス、およびガイドラインへの準拠という点で、関連するコードが期待に込んでいるかどうかを継続してモニタリングします。作業が正しく完了したことが客観的に保証されます。

Development Testing Platform は、ユーザー受け入れテストのスケジューリング、トラッキング、および文書化も補助します。ユーザー受け入れテストは、手動テスト インターフェイスを使って完了することができます。この手動テスト インターフェイスは、手動テスト プロセスに一貫性と再現性を加えます。さらに、過去のリリースの手動受け入れテストでカバーされた機能にコードの変更が影響を与えた可能性がある場合、開発チームにそのことが警告され、受け入れテストを再び実行する必要があることが通知されます。

---

## タスクの自動分配

計画されたタスクは、ビジネス ルールまたはリーダーなどによって指名された開発メンバーに自動的に分配されます。タスクは、Development Testing Platform Web インターフェイスからも、開発者が主に作業する IDE から参照できます。そのため、開発メンバーは正確な計画に簡単かつ頻繁にアクセスできます。全体的なプロジェクト目標、目標を達成するために完了しなければならない自分のタスク、および各タスクの期限を把握できます。

---

## タスクの自動モニタリング

割り当てられたタスクの作業を開始する準備ができたなら、チーム メンバーはタスクのステータスを「進行中」に変更し、通常と同じようにタスクの作業を行います。タスクが完了したらステータスを「完了」に変更します。チーム メンバーのワークフローを妨げることなく、タスクの分配テクノロジーは裏側で動作し、どのファイルがタスクに関連しているか、ファイルの作業にどのくらいの時間がかかるかをトラッキングします。これは変更ベースのテストにとって重要です。また、モニタリングされた結果から、開発チームは各タスクに設定された作業時間と実際に作業にかかった時間とを比較して、イテレーション全体の見積り時間を必要に応じて更新できます。

---

## 柔軟性の高いタスクの追加と見積りの変更

タスクを割り当てられたチーム メンバーは、さらに作業が必要だと判断した場合、柔軟に予測時間を更新したり新しいタスクを作成することができます。ソフトウェア開発ビジネスの特性として、多くの場合、具体的なタスクの作業時間を最初に予測することは不可能です。なぜなら、誰も予見できない隠れた作業が常に存在するからです。チーム メンバーがコードを詳しく調べ始めるまで、そのような作業は発見されません。

そのような場合、新しい作業が追加または提案されたことがマネージャーに直ちに通知されます。マネージャーは柔軟に新しいタスクを作り出してメンバーに分配できます。そのため、さらに優れた正確な情報がマネージャー側にフィードバックされます。

# DTP のインストール

このセクションの内容：

- システム要件
- Development Testing Platform クイック インストール ガイド

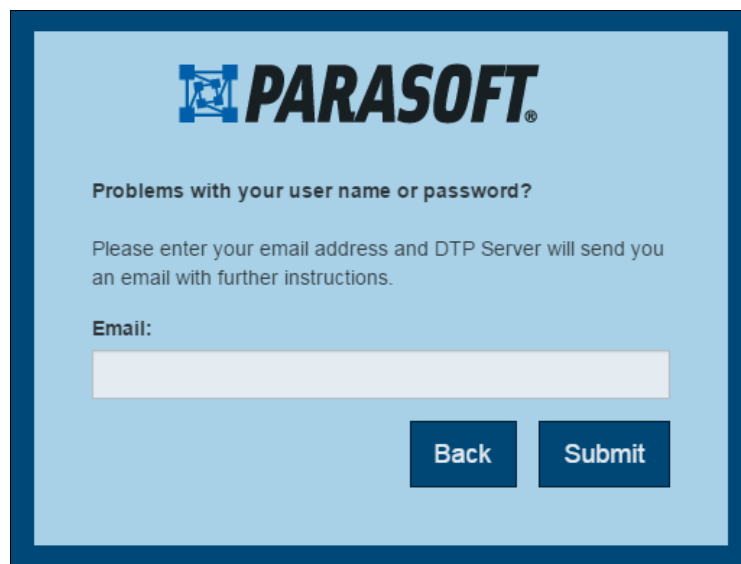
## パスワードの取得

Development Testing Platform へのログインで問題が発生している場合、以下の操作を行います。

1. ログイン ページの [ アカウントにアクセスできない場合 ] をクリックします。



2. 表示されたダイアログで、自分のユーザー プロファイルに関連付けられた電子メール アドレスを入力し、[ サブミット ] をクリックします。



3. 入力したアドレスに、アカウントへのアクセス方法を説明した電子メールが送信されます。

# Project Center の用語

次の表は、Project Center でのプロジェクト管理に関係する用語とその説明です。これらの用語はこのマニュアル全体を通して使用されます。

**注意：**組織や開発チームによっては、これらの用語をカスタマイズして使用している場合があります。

用語	説明
プロジェクト	<p>アプリケーションの特定のリリース（たとえば MyApplication 2.0 など）。プロジェクトはエンティティの最上位レベルです。</p> <p>プロジェクトには、開発チーム メンバー、実装する要求、達成するタスク、修正する欠陥 / 改善、およびイテレーション中のグループ タスクが含まれます。</p>
要求	<p>リリースに入れる予定の機能についてユーザー レベルで説明したもの。</p> <p>プロジェクトは、プロジェクト ライフサイクル中に実装する要求のリストから開始します。要求は、スプレッド シート文書 (xls ファイル) で定義して Project Center にインポートすることができます。</p>
欠陥	<p>顧客、開発者、QA、または他のチーム メンバーによって発見されたアプリケーションのバグまたは問題のレポート (PR)。</p> <p>ほとんどの場合、バグはバグ トラッキング システム (BTS) に格納されます。Project Center は、バグ トラッキング システムと統合して Development Testing Platform の機能を強化します。</p>
改善	<p>既存のアプリケーションに対する改善要求またはフィーチャー リクエスト (FR)。</p> <p>ほとんどの改善 は顧客、開発者、または QA によって要求されます。改善は BTS または要求管理システム (RMS) に格納されます。</p>
タスク	<p>プロジェクト チーム メンバーが割り当てられた作業の単位。</p> <p>タスクの例には、機能の実装、要求の実装、バグの修正が含まれます。ドキュメントの参照、新技術の調査、テスト インフラのセットアップといった作業もタスクになりえます。</p> <p>タスクはイテレーションと呼ばれる単位にまとめられます。</p>
イテレーション	<p>プロジェクトの 1 イテレーション。プロジェクトは多くのイテレーションを持ちます。</p> <p>イテレーション は、関連付けられたタスクのリストを持ちます。関連付けられたすべてのタスクが完了したときに イテレーション は完了します。</p>

# チームメンバーにとってのメリット

このセクションでは、Project Center のプロジェクト管理機能を利用することで各チームのメンバーにどのようなメリットがあるかを説明します。

- マネージャーによる Project Center の利用
- 開発者による Project Center の利用
- QA チームによる Project Center の利用



---

# マネージャーによる Project Center の利用

Project Center はマネージャーが以下の作業を行うのを支援します。

- プロジェクトの計画
- プロジェクト進捗のトラッキング
- 終了したプロジェクトの分析

---

## プロジェクトの計画

開発チームは、顧客要求を満たしてプロジェクトのリリース日に間に合うよう計画を作成することによって、プロジェクトを管理できます。Project Center の多彩な機能は、マネージャーが計画目標を達成するために役立ちます。プロジェクトの計画立案には次の業務が含まれます。

- 作業のスケジュールと配分

---

## 作業のスケジュールと配分

Project Center は、顧客要求と最終期限の両方に基づいてプロジェクト作業を計画するのに役立ちます。

マネージャーは、開発チームのサポートを受けながら、要求を満たすための作業タスクを作成します。タスクごとに、コストおよび作業完了までに必要な期間を見積ります。次に、タスクをスケジューリングしてイテレーションにします。

以下のどちらの状況も考えられます。

- イテレーションに最終期限が設定されている。そのため、最終期限を守るために必要な人数の開発チームにタスクを割り当てる。
- タスクを開発チームに割り当て、各タスクの作業時間を見積る。イテレーションの最終期限は自動的に計算される。

作成されたタスクは、タスクを割り当てられたチームメンバーに自動的に配分されます。チームメンバーは Web ページまたは IDE から自分のタスクに簡単にアクセスすることができます。

---

## プロジェクト進捗のトラッキング

プロジェクト ライフサイクルの全過程において、マネージャーは簡単にプロジェクト進捗をモニタリングし、プロジェクトの進捗を管理できます。マネージャーは「タスク クリープ グラフ」という各イテレーションの進捗をグラフィカルに表示するグラフを使って、開発ワークフローをすぐに確認することができます。このグラフの詳細については 198 ページの「タスク クリープ グラフ」を参照してください。

作業進捗をモニタリングすることで、残りのタスクの数を確認したり、最終期限までに完了しなければならぬ作業のボリュームを確認できます。その際、最終期限に間に合わせるためにタスクの再ス

スケジュールが必要かどうかを迅速に判断できます。たとえば、優先度の低いタスクを別のイテレーションに移動したり最終期限を延長したりする必要があるかを判断できます。

また、開発者の作業効率、時間のかかっているアクティビティ、効率アップが可能な分野などを分析し、最終的に開発の効率化に向けて適切な判断を下すことができます。

最終的な製品の品質を保証するために、マネージャーはプロジェクトが完了する前に適切な受け入れテストの実行をスケジュールできます。

図 レポート

反復数: 4 タスク数: 53

	ID	反復名	タスク数				予定	
			合計	Comp.	Prog.	オープン	開始日	作業時間
1	31	Milestone 3: Advanced functionality implementation and release	11	1	1	9	2012-09-26	60d
2	30	Milestone 2: Basic functionality implementation	12	0	0	12	2012-08-26	54d
3	29	Milestone 1: Architecture design and core framework implementation	14	8	2	4	2012-07-26	54d
4	8	フェーズ1	3	1	2	0	2010-10-08	29d
5		[どの反復にも属していないタスク]	13	5	0	8		
合計			53	15	5	33		197d

## 終了したプロジェクトの分析

プロジェクトの履歴データを利用すると、開発プロセス全体を検証して以下のような疑問に対する回答を得ることができます。

- どこがボトルネックとなったのか
- 最も時間がかかったアクティビティは何か
- 予測からどの程度のずれが発生したか
- 予測からのずれが最も大きかったチームメンバーは誰でタスクは何か
- 将来の開発プロセスを効率化するために何を改善できるか

---

# 開発者による Project Center の利用

Project Center は以下の機能によって開発者を支援します。

- 割り当てと期限の提示
- 履歴データの提示

---

## 割り当てと期限の提示

Project Center は、開発者に的確な攻略プランを提示します。開発者はプロジェクトの全体的なゴールを認識し、自分が完了すべき割り当て / タスクや、各割り当て / タスクの最終期限を知ることができます。

タスクが完了したら、開発者は QA チームにテスト シナリオを提供し、実装した機能に関する誤解を防ぐことができます（機能が顧客またはマネージャーからの要求に明確に定義されていない場合）。

---

## 履歴データの提示

開発者は自分が作業したタスクなどの履歴データとともに、各タスクに関連するソース コードや実行されたテストなどを参照できます。

ソース コードがモニターされていることは、開発者にとって大きな利点をもたらします。たとえば、機能を実装してしばらく期間が経った後で、修正の必要が発生するかもしれません。あるいは、バグを修正しなければならない場合もあるかもしれません。Project Center に ソースコードの登録の履歴が残っていれば、将来の変更やバグ修正が容易になります。

Project Center は開発者に自動テストおよび手動テストを提供します。テストはプロジェクト成果物（要求、欠陥 / 改善、タスク）単位で実行できます。

---

# QA チームによる Project Center の利用

Project Center は、QA チームがプロジェクト品質を保証するのを支援します。

---

## プロジェクト品質の保証

Project Center レポートは、貴重な情報を QA チームに提供します。

- 開発の進捗
- プロジェクト成果物（要求、タスク、欠陥 / 改善）のステータス。最近実装 / 修正され、テストが可能になったものは何か？

QA チームがテストを完了すると、ステータスがレポートされ、何を修正する必要があるかを開発者と QA の両者が知ることができます。

プロジェクトの完了にあたって、マネージャーは受け入れテストをスケジューリングし、プロジェクトの品質を保証するためにテストする必要のある項目を QA チームに伝えることができます。このような連携の結果として、製品の品質が向上し、市場への投入までの期間を短縮できます。

# 一般的なユース ケースとワークフロー

この一連のセクションでは、ソフトウェア開発ライフ サイクル (SDLC) 全体にわたって Project Center を利用して開発プロジェクトを管理する方法について説明します。以下のプロジェクト開発段階について説明し、一般的な開発プロジェクト ワークフローを示します。

- Project Center でのプロジェクト管理：一般的なケース
- プロジェクト計画
- 進行中のプロジェクト
- 開発プロセスの改善

ソフトウェア開発プロセスは組織によって違いがあるため、説明されているすべてのステップがお客様の作業体制にとって適切とはかぎりません。たとえば、要求仕様はまったく使用していないがバグ トラッキング システムとタスクは使用しているのであれば、要求に関わるステップをスキップすべきです。

電子署名ワークフローを構成する方法については 241 ページの「ワークフロー、権限、承認プロセスの構成」を参照してください。

# Project Center でのプロジェクト管理： 一般的なケース

Project Center は柔軟が高い製品です。どのように使用するかをユーザーが決定します。どのプロジェクト コンポーネント（要求、タスク、欠陥 / 改善）に対して作業するかを選択します。また、どの程度徹底して開発プロセスを解析し、管理するかを選択します。つまり、ソース コードのモニタリングや自動 / 手動テストを行う範囲を選択します。

次の表は、Project Center を利用してプロジェクトを管理する基本的な方法の一覧です。

ケース	メリット
要求だけを処理する	<ul style="list-style-type: none"> <li>• 要求をスプレッドシート (xls ファイル) に定義し、後で Project Center にインポートできる。</li> <li>• ソース コードおよびソース コードのリビジョンを要求ごとにモニターできる。</li> <li>• テスト (自動および手動) を作成、スケジュール、モニターし、製品の品質を保証できる。</li> <li>• [ 要求ステータス ] レポートを参照し、残りの要求の数、実装済みの要求の数、未テストまたはテストが失敗した要求に関する概要を確認できる。</li> <li>• 要求ごとのソース コードの概要を参照することで、どの要求の実装が複雑かを判断できる。</li> </ul>
欠陥 / 改善だけを処理する	<p>Project Center には以下の機能があります。</p> <ul style="list-style-type: none"> <li>• バグ トラッキング システムと統合し、必要に応じてその機能を拡張する。</li> <li>• PR/FR に関連したソース コードの変更を追跡する。</li> <li>• PR/FR に対して行われた自動および手動テストのステータスをモニターする。</li> <li>• ユーザーのシステム中の PR/FR のステータスをレポートし、変更をモニタリングする。</li> <li>• テスト結果を PR/FR に関連付け、修正または再テストの必要があるものを通知する。</li> <li>• PR/FR の修正完了時にテストを実行することを開発者 / QA チームに義務付ける。</li> </ul>

ケース	メリット
タスクだけを処理する	<p>Project Center には以下の機能があります。</p> <ul style="list-style-type: none"> <li>• 作業を小さく分割し、チームメンバーにタスクを割り当てる。</li> <li>• タスクを作成または修正したときに、すべての関係者に E-mail で通知を送る。</li> <li>• タスクごとに開発コスト（予定時間など）を見積り、プロジェクトを効果的に計画して進捗をモニターする。</li> <li>• タスクをイテレーションにグループ化することによって計画を容易にし、また各イテレーションが最終期限に間に合うよう順調に推移しているかを系統的にモニターする。</li> <li>• MS Project と統合してより効率的にタスクを管理する。</li> <li>• プロジェクトの完了に必要な時間を推測し、実現可能な最終期限を予測する。</li> <li>• 開発の進捗やテストステータスのモニターを可能にする。</li> </ul>
すべてのプロジェクト成果物、または上記のケースの任意の組み合わせを処理する	<ul style="list-style-type: none"> <li>• 上記のケースの組み合わせによって異なります。</li> </ul>

上記のシナリオも含め、Project Center でプロジェクトを管理する方法の詳細については 17 ページの「プロジェクト計画」を参照してください。

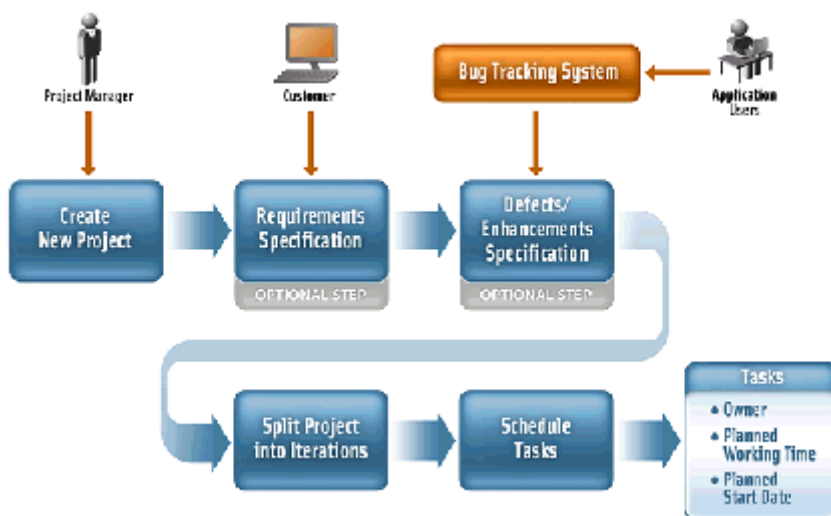
# プロジェクト計画

## ソフトウェア開発ライフサイクル全体でのプロジェクト管理：概要

Development Testing Platform を使用するプロジェクト計画段階は、一般的に以下のステップを含みます。

1. プロジェクトの 新規作成
2. 要求の指定
3. 欠陥 / 改善の指定
4. バックログのインポート
5. イテレーションの計画
6. イテレーション内のタスク管理

図 1: プロジェクト計画



## プロジェクトの 新規作成

要求の計画、タスクのスケジューリング、レポートのモニタリングなど、Project Center のほとんどのアクティビティは、すべての成果物をまとめる特定のプロジェクトのコンテキストで実行されます。プロジェクトを作成する際、いくつかの事項を考慮する必要があります。

- **名前** プロジェクトの命名規則は、製品スケジュールを反映したものにするべきです。複数のリリースは、通常は個別のプロジェクトで管理されます。Parasoft は [アプリケーション名 リリース\_バージョン] という規則を使用します (例: DTP 4.7)。どのような規則を採用する場合でも、すべてのリリースで一貫性を保つべきです。



- **チームのメンバー** まだ存在していない場合、プロジェクトに関連付けるチームを作成します。

プロジェクトを新規作成する方法については 230 ページの「プロジェクトの作成と構成」を参照してください。

## 実施する作業の定義

Development Testing Platform は、通常作成されるあらゆるプロジェクト コンポーネントおよび成果物を管理できますが、それにはまず、どんな作業を実施するのかを計画する必要があります。検討すべき一般的なワークフローは以下のとおりです。

1. 改善 / 欠陥のために要求を作成する
2. 改善 / 欠陥のためにコードを作成する
3. コードをテストする

## 要求の指定

プロジェクトの機能要求を Project Center に追加します。通常、要求は定義されるか顧客から収集され、手書きまたは .xls ファイルなどの電子フォーマットで文書化されます。その後、要求を Project Center に手動で 1 つずつ入力するか (79 ページの「要求の作成と管理」を参照)、または .xls や CSV ファイルからインポートできます。

### 要求のサンプル

要求						
検索の詳細: プロジェクト: PolicyCdemo				保存済み検索: --カスタマイズ済み--		
29 アイテム						
<input type="checkbox"/> タスクとしてスケジュール <input type="checkbox"/> 印刷 <input type="checkbox"/> 一括更新 <input type="checkbox"/> 再スキャン						
<input type="checkbox"/>	ID	名前	説明	ランク	ステータス	値
<input type="checkbox"/>	93	Easy user registration in the shop	It should be comfortable and almost seamless for th...	50000 - M	作業中	91
<input type="checkbox"/>	25	本体の組み上げ		50000 - M	作業中	49
<input type="checkbox"/>	92	Extra security needed	We should use all state-of-the-art technologies to ...	50000 - M	作業中	91
<input type="checkbox"/>	95	Browsing through the shop products	It should be easy and intuitive for customers to br...	50000 - M	作業中	
<input type="checkbox"/>	27	音声認識と視覚性能		50000 - M	承認待ち	26
<input type="checkbox"/>	94	User profile page	Each user, after getting registered has his persona...	50000 - M	新規	91
<input type="checkbox"/>	26	口語指示に従う		50000 - M	新規	49
<input type="checkbox"/>	28	命令の実行(走る、持つ、座る)		50000 - M	新規	26
<input type="checkbox"/>	91	User registration and profile	This is a top level requirement representing functi...	50000 - M	新規	

## 欠陥 / 改善の指定

通常、プロジェクトが以前のバージョンから継続している場合、バグ トラッキング システム (BTS) に欠陥 / 改善がすでに登録されています。一般的に、BTS は開発プロジェクトで使用され、プロジェクトの関係者、ユーザー、および開発者はアプリケーションで発見した欠陥を登録したり、改善のアイデア (フィーチャー リクエスト) を登録することができます。Project Center は多くの BTS と統合でき、欠陥 / 改善とプロジェクトを結び付けます。

## Development Testing Platform で BTS の欠陥 / 改善を検索する

General

ID(s):	<input type="text"/>	Summary:	<input type="text"/>
Type:	any <input type="button" value="v"/>	Req. Summary:	<input type="text"/>
Task Name:	<input type="text"/>	Scheduled as Task?	Any <input type="button" value="v"/>
Projects:	Concerto 10 <input type="button" value="v"/> Concerto 3.5+ <input type="button" value="v"/> Concerto 4+ <input type="button" value="v"/> Concerto 4.0 Acceptance Testin <input type="button" value="v"/>	Resolved:	Any <input type="button" value="v"/>

### Advanced

Defect Tracking System:

- Bugzilla
- GRS
- Tool
- ClearQuest
- JIRA
- CSV
- My Custom BTS
- Test Director

## バックログのインポート

プロジェクトをビルドした後は、要求、欠陥または改善、タスクなどのバックログ データをインポートする必要があります。バックログを Development Testing Platform にインポートすると、以下のメリットがあります。

- 1つの場所で一元的に管理でき、管理が容易になる。
- 複数のリリースにわたって類似したタスクを追跡できる。

### 用語についての注意

タスクをインポートする場合、タスク タイプは「要求の実装」となり、まぎらわしい可能性があります。

## 要求のインポート

1. [要求] > [インポート] をクリックします。
2. ドロップダウン メニューからプロジェクトを選択します。
3. 要求ファイルを選択し、[ファイルのアップロード] をクリックします。

4. ファイルのアップロードが完了したら、[ 選択されたファイルからのカラム名 ] ドロップダウンメニューから適切なプロパティを選択し、要求のマッピングを設定します。

Step 3: Configure Mapping

Imported fields: *	Concerto Requirement Fields	Column Names from Selected File
	Name	---
	Original ID	---
	Description	ID Priority Description Status
	Rank	---
	Status	---
	Due Date	---
	Owner(s)	---
	Story Points	---
	Parent Requirement ID	---
	Field Label	---

5. [ 次へ ] をクリックし、インポートされた要求を確認して [ サブミット ] をクリックします。

## 欠陥 / 改善のインポート

1. [ 欠陥 ] > [ インポート ] をクリックします。
2. ドロップダウンメニューからプロジェクトを選択します。
3. 要求ファイルを選択し、[ ファイルのアップロード ] をクリックします。
4. ファイルのアップロードが完了したら、[ 選択されたファイルからのカラム名 ] ドロップダウンメニューから適切なプロパティを選択し、要求のマッピングを設定します。
5. [ 次へ ] をクリックし、インポートされた要求を確認して [ サブミット ] をクリックします。

## ファイルからの欠陥のインポート

1. .csv、.xls、または .xlsx ファイルから Development Testing Platform のプロジェクトに内部的な欠陥をインポートすることができます。ファイルから欠陥をインポートすると、Development Testing Platform はそれらの欠陥を内部的な欠陥として認識し、より多くのプロパティを定義できます。

**注意：**別の方法として、BTS から外部的な欠陥をインポートすることもできます。外部的な欠陥をインポートすると、Development Testing Platform は欠陥の履歴およびステータスのほかに、BTS によって定義されたプロパティを認識します。詳細については「330 ページの「CSV からの BTS データのインポート」を参照してください。

1. 欠陥をリストした CSV ファイルおよびプロパティを作成します。

例 ::

	A	B	C	D	E	F	G	H	I	J	K
1	id	Name	Description	Rank	Status	Parent	Owner	Deadline	Implementation	Progress	
2	142	Test Requirement		50000	Review	1191			100		
3	3620	Support for Visa	We should implement the support for	50000	Review	3618			0		
4	3618	Support for credit cards	Afer purchasing the product, useer	50000	Review				0		
5	3619	Support for AmEx	We should implement the support for	50000	Review	3618			0		
6	3616	Nice online reports	We should have reports presenting	50000	Review	3614			0		
7	3617	Flexible look and feel	Each user of our shop shuld be able	50000	Review	3614			0		
8	3613	Functionality of giving rating	On view product details page there	50000	Review	3609			0		
9	1135	test-requirement (throw away when done)		50000	Review	1191	jhicken	#####	0		
10	3612	Functionality of giving feedb	Provide a link in thefooter, next to	50000	Review	3609			0		
11	3615	Stock Control	It should be easy for our shop	50000	Review	3614			0		
12	3614	Tools for administration	It is a top level requirement for	50000	Review				0		
13	1191	Test List		50000	Review				50		
14	3609	Feedback mechanism	User should be able to send us	50000	Review				0		
15	3608	Secure payment methods	This is a top level requirement	50000	Review	3606			0		

2. [欠陥] > [インポート] を選択します。

**注意:** CSV ファイルに各国語の文字が含まれている場合 (たとえば欠陥名などで)、ファイルを Development Testing Platform にインポートする前に、UTF-8 エンコーディングで保存されていることを確認してください。

3. 欠陥と関連付けるプロジェクトを選択します。
4. [追加] をクリックして欠陥を含むファイルを参照します。
5. [アップロード] をクリックし、CSV ファイルのフィールドを Development Testing Platform のフィールドにマッピングします。必須のフィールドは名前だけです。
6. [次へ] をクリックし、インポートされた欠陥を確認します。
7. [サブミット] をクリックして終了するか、[戻る] をクリックしてフィールドのマッピングまたは別のファイルのインポートに戻ります。

## タスクのインポート

1. [タスク] > [インポート] をクリックします。
2. ドロップダウンメニューからプロジェクトを選択します。
3. 要求ファイルを選択し、[ファイルのアップロード] をクリックします。
4. ファイルのアップロードが完了したら、[選択されたファイルからのカラム名] ドロップダウンメニューから適切なプロパティを選択し、要求のマッピングを設定します。
5. [次へ] をクリックし、インポートされたタスクを確認して [サブミット] をクリックします。

## 反復的タスク

プロジェクト チーム メンバーは、各イテレーションまたはリリースで繰り返される反復的なタスクを行うことがよくあります。これらのタスクを追跡すると、生産性向上の余地を発見するのに役立つ場合があります。そのための最良の方法は、複数の反復的タスクを 1 つの要求にまとめ、反復的タ

スクのカテゴリごとにサブ要求を作成することです。その後、それらの要求を確認し、プロジェクトのライフサイクル全体で、どの程度の時間が反復的タスクに消費されたかを追跡します。よくある反復的タスクのカテゴリとしては、以下が挙げられます。

- 管理タスク
- 開発者タスク
- リリースタスク（ソースコードのブランチの作成、インストーラーのビルド、リリースノート作成、マーケティングとの連絡、ダウンロードサイトへのインストーラーのアップロードなど）
- バッファ（予期しないタスクのためにあらかじめ見積られた時間）
- サポート（開発者がサポートアクティビティを実行する場合）

欠陥 / 改善の操作については、106 ページの「要求と欠陥 / 改善のタスク化」を参照してください。

## イテレーションによる作業のスケジュール

要求、欠陥、改善を収集すると、必要なタスクの全体像が見えてきます。そしてマネージャーはタスクをイテレーションに分割することができます。アジャイル開発において、プロジェクトをイテレーションまたは「スプリント」に分割することは、作業をスケジュールする理にかなった方法です。

### 新規イテレーション

The screenshot shows a web form for creating a new iteration. It has the following sections and fields:

- Header:** '新規反復' (New Iteration) and '全般' (General).
- Form Fields:**
  - プロジェクト: PolicyCdemo (dropdown)
  - 名前: (text input)
  - 説明: (text area)
- Schedule Section:**
  - スケジュール済みリソース (Scheduled Resources)
  - Dev.#: 1.0 (text input)
  - ゼロ キャパシティ 0d (text input)
- Time Section:**
  - 作業時間 (Work Time)
  - 予定 (Schedule)
  - 開始日: (calendar icon)
  - 終了日: (calendar icon)
- Action:** 作成 (Create) button

イテレーションの開始日と終了日は、ここですぐに指定することも、タスクを追加した後で指定することもできます。[開始日]と[終了日]のフィールドの隣にあるカレンダーのアイコンをクリックして指定します（「イテレーション内のタスク管理」を参照）。

イテレーションの作成方法については 112 ページの「イテレーションへのプロジェクトの分割」を参照してください。

---

## イテレーションの計画

イテレーションタスクは複数の場所で計画できます。

- [Planning Iterations] ページ。114 ページの「イテレーションの計画」を参照。
- 要求と欠陥 / 改善のリスト。106 ページの「要求と欠陥 / 改善のタスク化」を参照。
- [イテレーションの編集] ページ。137 ページの「イテレーションの編集」を参照。
- [新規タスク] ページ。123 ページの「タスクの新規作成」を参照。
- MS Project。MS Project でイテレーション タスクを開き、MS Project のスケジュール計画を利用できます。

イテレーションの終了日の予定に関しては、通常、以下のいずれかのケースが当てはまります。

- イテレーションの最終期限が設定されていて、特定の機能（タスク）を最終期限までに実装する必要がある。
- イテレーションの最終期限は設定されていないが、特定の機能（タスク）を実装する必要がある。

どちらの場合も、マネージャーは適切なイテレーションにタスクを割り当てる必要があります。詳細については 26 ページの「イテレーションの終了予定日の変更」を参照してください。

---

## イテレーションの作成

イテレーションには、組織の開発プロセスの構成に基づいた命名規則に従って、固有の名前を付ける必要があります。次の規約が推奨されます。

- チームごとに 1 つのイテレーションを作成する
- マイルストーンでイテレーションをグループ化する（つまり イテレーションはプロジェクト - マイルストーン - チーム / グループとなる）
- 例：DTP 4.6 - July - DEV-SC

イテレーションの計画と作成にあたっては、利用可能なリソースおよびロードマップを考慮することを推奨します。

- このイテレーションのグループにはいくつの FTE があるか？
- このマイルストーンの開始日および終了日はいつか？

---

## タスクの指定

プロジェクト計画を効率的に進めてプロジェクトの進捗をモニタリングするには、すべての開発アクティビティをタスクに分割し、個々のタスクをスケジュールリングする必要があります。タスクには、要求の実装、欠陥の修正、開発環境の準備、システムの設計など、さまざまなアクティビティが含まれます。

Project Center には、以下の定義済みのタスク タイプがあります。

- **スタンドアロン**：他のすべてのタスクに使用。
- **設計 / 計画**：アプリケーションの設計に関連するタスクに使用。
- **要求の実装**：要求の作成に関連するタスクに使用。

- **欠陥 / 改善の修正** : 欠陥または改善の修正に関連するタスクに使用。
- **品質** :

タスクをインポートするほかに、タスクを新規作成できます。

[ タスク ] ドロップダウン メニューの [ 新規 ] をクリックします。

1. タスクの情報およびメタデータを入力します。
2. [ 作成 ] をクリックします。

タスク管理画面からタスクを作成することもできます。

1. [ タスク ] ドロップダウン メニューの [ 管理 ] をクリックします。
2. [ 新規タスク ] ボタンをクリックします。
3. タスクの情報およびメタデータを入力します。
4. [ 作成 ] をクリックします。

詳細については 25 ページの「イテレーションへのタスクの追加」を参照してください。

要求および欠陥 / 改善を自動生成された条件付きのタスクとしてスケジュールできます。詳細については 106 ページの「要求と欠陥 / 改善のタスクとしてのスケジューリング」を参照してください。

リソースを消費する重要なアクティビティがプロジェクトで計画されている場合、そのようなアクティビティに対してタスクを作成することを推奨します。特に以下のものに対して推奨します。

- **要求** : まず顧客またはマネージャーがアプリケーション ユーザー言語で要求を指定したら、次に開発言語を使って要求を開発タスクに分割します。通常、この作業はマネージャーまたはチーム リーダーが行います。一般的に、1 つの要求が複数の開発タスクに分割されます。

要求をタスクとしてスケジュールする方法の詳細については 106 ページの「要求と欠陥 / 改善のタスク化」を参照してください。

- **欠陥 / 改善** : バグ トラッキング システムの欠陥 / 改善の中で、マネージャーがプロジェクト ライフサイクル中に修正 / 実装すると決定したものは、それぞれ 1 つのタスクとしてスケジュールする必要があります。欠陥 / 改善をタスクとしてスケジュールする方法の詳細については 106 ページの「要求と欠陥 / 改善のタスク化」を参照してください。
- **その他** : プロジェクトの開発中には、要求の実装や欠陥の修正を含まないアクティビティも発生します。たとえば、アプリケーションの設計やインフラ ( 環境 ) の準備、技術の習得、テスト、デプロイメントなどです。マネージャーは、プロジェクト計画フェーズでこれらのアクティビティのタスクを作成する必要があります。25 ページの「イテレーション内のタスク管理」を参照してください。

MS Project のタスクを Development Testing Platform のイテレーションに送信して同期することができます。詳細については 207 ページの「Microsoft Project の統合」を参照してください。

**注意** : タスクを新規作成すると、関係者にメールが送信されます。詳細については 389 ページの「タスクと欠陥の電子メール通知」を参照してください。

プロジェクトのすべての作業をタスクに分割したら、累計作業見積り時間を持つオーナーに各タスクを割り当てます。この情報に基づいて、マネージャーはイテレーションを効率的に計画し、チームメンバーにタスクを割り当て、進捗をモニタリングし、プロジェクトの完了日を予測できます。

## イテレーション内のタスク管理

マネージャーはプロジェクトの進行中にタスクとイテレーションを変更できます。

- イテレーションへのタスクの追加
- イテレーションの終了予定日の変更

### イテレーションへのタスクの追加

1. 要求、欠陥、改善、反復的タスクを検索します。\*
2. イテレーションに所属するものを決定します。
  - ロードマップおよび上位レベルの計画に従います。
3. 計画するイテレーションのタイプを決定します。
  - 開発、テスト、ドキュメンテーション、一般的なプロジェクト管理など
  - タスクの決定
4. バックログの各アイテムについて作業を見積ります。
  - ストーリーポイントを使用している場合、見積りを自動入力するために使用できません。
  - チームで相談する必要がある場合もあります。
5. タスクを誰に割り当てるかを決定します。
  - タスクに適合する経験およびスキルセット
  - [タスク オーナー別の作業] ページでタスクの配分を確認し、作業負荷に基づいてタスクを割り当てます。
    1. Project Center のメインメニューで [イテレーション] をクリックします。
    2. [ID] の番号をクリックしてイテレーションの詳細ページを表示します。
    3. [タスク オーナー別の作業] タブをクリックします。

#### \* 保存された検索条件の使用

成果物を検索し、イテレーションに追加する際に、Development Testing Platform の検索機能を活用することを推奨します。検索を保存すると、検索条件によってカスタマイズされたレポートのような働きをします。以下のような検索条件がよく使用されます。

- 自分が管理しているユーザーごとの全タスクのリスト
- スケジュールされていない要求のリスト
- オープン中の欠陥のリスト
- スケジュールされていないドキュメンテーション PR/FR のリスト
- テストされていないタスクのリスト

タスク、要求、欠陥の名前または ID など、探している対象に関連するキーワードが分かる場合は、[クイック検索] バーを使用してすばやく検索対象を見つけることができます。



別のイテレーションにタスクを移動する方法については、137 ページの「イテレーションの編集」を参照してください。

---

## イテレーションの終了予定日の変更

マネージャーはイテレーションの終了予定日を確認して、以下の場所で日にちを変更することができます。

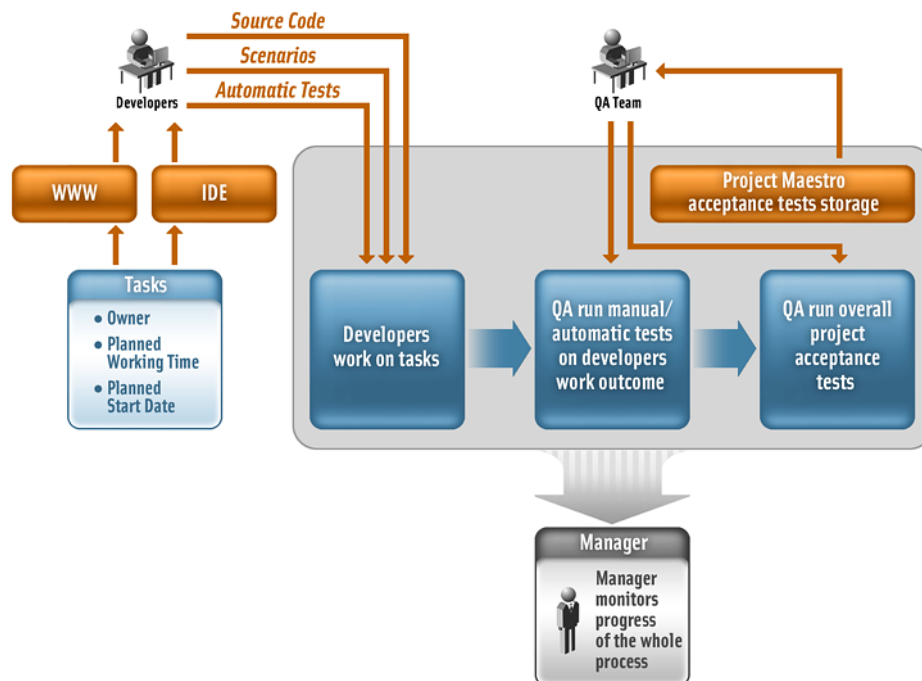
- [イテレーションの編集] ページ：すべてのタスクの見積り作業時間の合計、およびイテレーション全体の推定終了日が表示されます。詳細については 137 ページの「イテレーションの編集」を参照してください。
- [プロジェクト ステータス] レポート：プロジェクト開発の全体的な進捗が表示されます。プロジェクトにおけるユーザーの現在のステータスが表示されます。リスクと乖離を管理することもできます。
- MS Project: Project Center と MS Project を統合すると、複雑なタスク管理操作が可能になります。マネージャーは MS Project で簡単にイテレーションを開いてタスクの編集（ユーザーの再割り当て、開始日の変更、タスクの追加や削除など）を行い、変更を Project Center にアップロードして推定完了日の変化を確認できます。

## 進行中のプロジェクト

通常、Development Testing Platform での作業の第二段階には次のステップが含まれます。この第二段階をモデル化したのが 図 4 です。

1. タスクに対する作業
2. プロジェクトのモニタリング
3. 予定された最終期限に間に合わせるためのタスク管理
4. イテレーションの品質保証
5. 受け入れテストの定義
6. 欠陥の修正
7. 結果のテスト

図 4: 作業中のプロジェクト



## タスクに対する作業

プロジェクトを完了するために必要なすべてのアクティビティに対してタスクを作成したら、開発とチームの共同作業を計画し、作業をモニタリングすることができます。通常、タスクに対する作業では以下のステップを実行する必要があります。

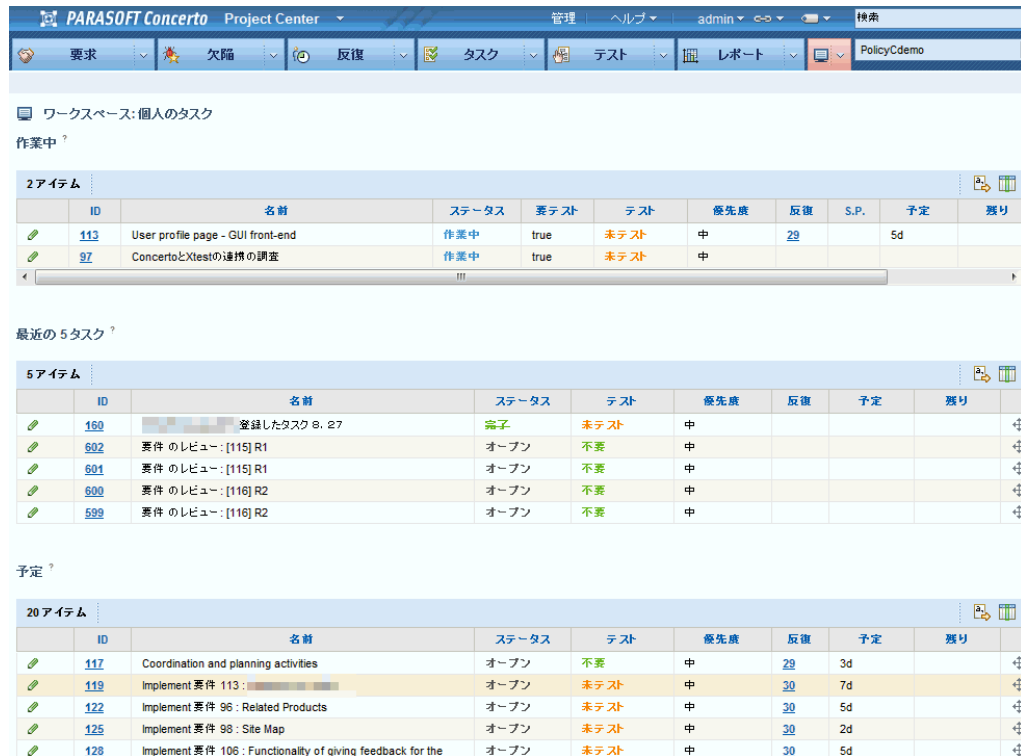
1. Project Center にログインしたら、[ マイ タスク ] のリストを参照します。これは自分に割り当てられたタスクのリストであり、以下の 3 種類に分類されます。

- 作業中
- 最近のタスク
- 予定



注意：[マイ タスク] タブ  をクリックしてこのページを表示することもできます。詳細については 47 ページの「ワークスペース / マイ タスク」を参照してください。

図 5: マイ タスク



ID	名前	ステータス	テスト	優先度	反復	S.P.	予定	残り
113	User profile page - GUI front-end	作業中	true	未テスト	中	29	5d	
97	ConcertoとXtestの連携の調査	作業中	true	未テスト	中			

ID	名前	ステータス	テスト	優先度	反復	予定	残り
160	登録したタスク 8, 27	完了	未テスト	中			
602	要件のレビュー: [115] R1	オープン	不要	中			
601	要件のレビュー: [115] R1	オープン	不要	中			
600	要件のレビュー: [116] R2	オープン	不要	中			
599	要件のレビュー: [116] R2	オープン	不要	中			

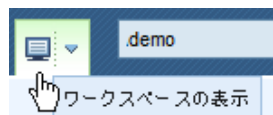
  

ID	名前	ステータス	テスト	優先度	反復	予定	残り
117	Coordination and planning activities	オープン	不要	中	29	3d	
119	Implement 要件 113: [115] R1	オープン	未テスト	中	30	7d	
122	Implement 要件 96: Related Products	オープン	未テスト	中	30	5d	
125	Implement 要件 98: Site Map	オープン	未テスト	中	30	2d	
128	Implement 要件 106: Functionality of giving feedback for the	オープン	未テスト	中	30	5d	

2. 簡略版のマイ タスク リストを参照したい場合は [マイ タスク] タブの矢印をクリックします。

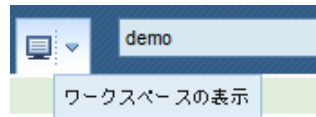
作業中のタスクが 1 個の場合、[マイ タスク] タブは緑で表示されます。

図 6: [マイ タスク] タブ (緑)



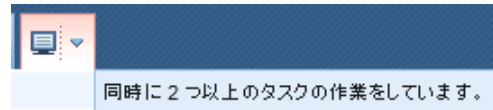
作業中のタスクがない場合、青で表示されます。

図 7: [マイ タスク] タブ (青)



作業中のタスクが 2 個以上ある場合、赤で表示されます。

図 8: [マイ タスク] タブ (赤)



**注意：** 詳細については 47 ページの「ワークスペース / マイ タスク」を参照してください。

3. 自分に割り当てられたタスクを参照できるほか、開始日と残り時間を参照できます。

図 9: タスクの表示 - 検索結果

ID	名前	ステータス	テスト	優先度	反復	オーナー	予定
25	要件 25 の実装: 本体の組み上げ	完了	成功	高	ⓘ		10d
26	要件 27 の実装: 音声認識	作業中	未テスト	中	ⓘ	kylie	9d
28	要件 28 の実装: 命令の実行(走る、持つ、座る)	作業中	未テスト	中	ⓘ		10d

- [ 検索 ] メニュー項目をクリックして自分に割り当てられたタスクを参照します。
  - タスクの検索ウィンドウが表示されます ( 図 10)。

- 検索条件の [ タスク オーナー ] フィールドで自分のログイン名を指定します。

図 10: 検索

全般

タスク ID: <input type="text"/>	タスク ステータス: <input type="text" value="オープン"/>
オリジナル ID: <input type="text"/>	<input type="text" value="作業中"/>
タスク名: <input type="text"/> <input type="checkbox"/> Not	タスク優先度: <input type="text" value="最高"/>
説明/コメント: <input type="text"/>	<input type="text" value="高"/>
	<input type="text" value="中"/>

タスクの関係

反復 ID: <input type="text"/> [検索]	タスク プロジェクト: <input type="text" value="Support2012"/>
反復内にある: <input type="text" value="任意"/>	<input type="text" value="Workflow-1"/>
要求 ID: <input type="text"/> [検索]	<input type="text" value="Workflow-2"/>
欠陥 ID: <input type="text"/> [検索]	<input type="text" value="テスト今井"/>
タグ: <input type="text"/> <input type="checkbox"/> Not	<input type="text" value="永田プロジェクト"/>
	タスク オーナー: <input type="text"/> <input type="checkbox"/> 空
	作成者: <input type="text"/>

タスクテスト

テスト ステータス: <input type="text" value="--"/>	(再)テストが必要: <input type="checkbox"/>
--	-------------------------------------

タイプ:

タスクの作成/変更



作成日: <input type="text"/>	修正日: <input type="text"/>
<input type="text"/>	<input type="text"/>
<input type="button" value="検索"/> <input type="button" value="消去"/>	




- Eclipse または Visual Studio を IDE として使用している場合、Parasoft Test のタスク アシスタントを使って IDE で直接タスクに対して作業することができます。

### タスクのクイック変更

タスクをすぐに修正したい場合、直接タスク リストからタスクを編集できます。

- 特定のタスクを編集モードにするには、[クイック変更] ボタンをクリックするか、そのタスクの行をダブルクリックします。そして必要な変更を行います。

<input type="checkbox"/>		ID	名前
<input type="checkbox"/>		35	要件 25 の実装: 本体の組み上げ
<input type="checkbox"/>			この行をクイック変更 [dblclick] 実装: 音声認識


<input type="checkbox"/>		ID	名前
<input type="checkbox"/>	 	35	要件 25 の実装: 本体の組み上げ
<input type="checkbox"/>		36	要件 27 の実装: 音声認識

- タスクへの変更をコミットするには、[コミット] ボタンをクリックするか、Enter キーを押します。
- タスクへの変更をキャンセルするには、[キャンセル] ボタンをクリックするか Esc キーを押します。

#### 4. 特定のタスクに対して作業します。

- 割り当てられたタスクを「タスクの開始時に作業中」としてマークします。この操作は次の場所で行うことができます。
  - ワークスペース :  
[マイ タスク] ページ。予定されたリストから作業中のグループにタスクをドラッグ アンド ドロップできます。鉛筆のアイコンをクリックするかタスク行をダブルクリックすると、タスクを編集したりステータスを作業中に変更することができます。詳細については 47 ページの「ワークスペース / マイ タスク」を参照してください。
  - [マイ タスク] ドロップダウン リスト :  
行をクリックしてタスクのステータスを切り替えます。たとえば、作業中のリストにあるタスクをクリックすると、ステータスを変更するためのドロップダウン リストが表示されます。詳細については 47 ページの「ワークスペース / マイ タスク」を参照してください。

#### 作業中

113	User profile page - GUI front-end	ステータスの変更	
97	ConcertoとXtestの連携の調査	オープン	 このタスクのステータスを変更
		完了	
		キャンセル	

最近の 5 タスク

- [タスクの編集] ページ：  
Development Testing Platform の任意のページからタスク ID をクリックすると、[タスクの編集] ページが表示されます。

図 11:[タスクの編集] ページ

タスクの編集: [36] 要件 27 の実装: 音声認識

全般 変更履歴 コード 506 自動テスト 0 シナリオ 0

ステータス: 作業中

残り時間: 5d 3h 58m 進捗: 522d 2h 11m (5803%)  
 実際の時間: 525d 6h 13m

新規コメントにスクロール

要求の実装 27 説明およびコメント (0) 添付 (0)

説明およびコメント  
 説明 応答

新規コメント:

添付  
 0 アイテム 追加

名前	説明	添付日
表示データなし		

保存

プロジェクト: PolicyCdemo  
 名前: 要件 27 の実装: 音声認識  
 反復 ID: 8 [検索] [追加] [リニュー]

優先度: 中

オーナー: kylie [表示] [閉去]  
 監督者: masuya [表示] [閉去]  
 作成者: admin

タイプ: 開発

テストが必要:

タグ: このエンティティに対して定義されたタグはありません  
 [タグの追加]

開始日: 2010-10-11  
 9d

- Parasoft タスク アシスタント IDE プラグイン中の [タスクの編集] ページ。タスク アシスタントの詳細については『Parasoft Test ユーザーズ ガイド』を参照してください。
- **重要!** Development Testing Platform は、各タスクの作業中ステータスの累積継続時間に基づいて [実際の時間] を計算します。作業が完了したのにタスクを「作業中」のままにした場合などに、[実際の時間] の正確さを誤って低下させることがあります。ステータスの更新ミスによる "ずれ" を修正する方法については 予測作業時間の変更 を参照してください。
- 特定のタスクに対して作業する間、作業をテストするために自動テストまたは手動テストを用意することができます。Project Center では、現在のタスクにテストを関連付けることができます。その後、Project Center のレポートによってタスクのテスト ステータスをモニタリングできます。さらに、そのタスクで要求を実装するか欠陥を修正する場合、タスクのテストは要求または欠陥に個別に関連付けられます。
- 自動または手動のテスト シナリオに対して作業する場合、ソース コードが Parasoft Source Scanner モニターによってモニタリングされます。ソース コードに変更が加えられ、表示されたテストに対する作業中にコードがコミットされる場合、影響があるファイルについての記録がテストに転記されます。さらに、この情報は影響がある要求に転記されます。

**注意：**

- タスクが変更されるたびに、変更の詳細についての電子メールが関係者に送信されます。詳細については 223 ページの「タスク、欠陥、要求の電子メール通知」を参照してください。
- 作業中のタスクが、統合されたバグ トラッキング システム (BTS) 中の課題に関連付けられている場合、Project Center でこのタスクのステータスが変更されるたびに、BTS 中の対応する課題のステータスも合わせて更新されます。また、Project Center はそのようなタスクについてのコメントを BTS 中の対応する課題に伝播します。詳細については 226 ページの「バグ トラッキング システムとの同期」を参照してください。
- タスクが作業中の場合、毎日 [ 残り時間 ] を更新する必要があります。その目的は、マネージャーとチーム メンバーに、タスクの予定完了日について常に最新の情報を提供するためです。[ 残り時間 ] を更新するには 2 つの方法があります。
  - タスクを編集して [ 残り時間 ] を更新します ( 図 12)。

図 12: タスクの編集 - 残り時間

タスクの編集: [36] 要件 27 の実装: 音声認識

全般 変更履歴 コード 506 自動テスト 0 シナリオ 0

ステータス: 作業中

残り時間:  乖離:

実際の時間: 525d 6h 13m 正しい書式: 1d, 1d 4h など

- Eclipse または Visual Studio IDE でタスク アシスタント プラグインを使用します。朝 IDE を起動すると、ダイアログが表示され、現在作業中のタスクの [ 残り時間 ] を更新するように求められます。

作業の進捗は以下の場所で参照できます。

- タスク アクティビティ レポート。詳細については 192 ページの「[ タスク アクティビティ ] レポート」を参照してください。
- イテレーションのステータス レポート。詳細については 174 ページの「[ イテレーションのステータス ] レポート」を参照してください。

**5. 特定のタスクを完了します。**

- 特定のタスクの作業中にファイルを変更した場合、次のいずれかの方法で、ファイルの変更をその特定のタスクに関連付けます。
  - **ソース管理システムから：**コードの変更をソース リポジトリにコミットするときに、コメントに特別な記号 @task を「@task #」の形式で追加します。# の部分でタスク ID を指定します。その後には通常のテキスト コメントが続きます。コミットしたこのコードを Parasoft Source Scanner が定期的にスキャンします。Parasoft Source Scanner は、ソース コードに対する変更について Project Center に通知し、コミットされたソース コードを適切なタスクに関連付けます。



- **タスク アシスタントまたはタスク エクスプローラーから** : Parasoft Test のタスク アシスタントまたはタスク エクスプローラーを使用している場合、変更されたファイルを Parasoft Test が自動的にトラッキングします。タスク アシスタントまたはタスク エクスプローラーでタスクが無効化されると、ソースの変更についての情報が自動的に Development Testing Platform に送られます。詳細については『Parasoft Test ユーザーズ ガイド』のプロジェクト タスクの作業についてのセクションを参照してください。
  - 自動テストでも手動テストでも、テストを実行することを強く推奨します。プロジェクト成果物のテスト ステータスは Project Status レポートで見ることができます。
    - プロジェクト成果物とテストを関連付ける方法については 70 ページの「Project Center 成果物 (欠陥、要求、タスク、テスト) の関連付け」を参照してください。
    - テスト ステータスの可視性については 173 ページの「プロジェクト ステータス レポート」を参照してください。
6. タスクが完了したら、Project Center ホーム ページまたは Eclipse/Visual Studio のタスク アシスタント プラグインから、タスクのステータスを「完了」に変更できます。また、タスクがキャンセルされる場合はタスクのステータスを「キャンセル」に変更できます。

#### 注意：

- タスクが変更されるたびに、変更の詳細についての電子メールが関係者に送信されます。詳細については 223 ページの「タスク、欠陥、要求の電子メール通知」を参照してください。
- Project Center は、特定のタスクに対して作業しているすべてのユーザーの活動をモニタリングします。タスクの計算アルゴリズムについては 227 ページの「作業中のタスクの経過時間 (実時間) の計算方法」を参照してください。

## プロジェクトのモニタリング

タスクの完了に向けて作業する間、ユーザーとマネージャーは自分の作業量と開発の進捗をモニタリングすることができます。Project Center を使用すると、以下の情報を確認することができます。

- 開発チームの各メンバーが現在どのタスクに対して作業しているか。
- 現在のイテレーションに必要なタスクをすべて完了するには何日必要か。必要な日数は、終了予定日より前にタスクを完了できる日数か。
- どんなソース コードがソース管理システムにチェックインされたか。最近作業したタスク、要求、およびバグのためにチェックインされたコード行は何行か。
- どんな自動テストと手動テストが作成されたか。レポートされたテストの実行ステータスは何か。
- イテレーションが開始してから新しいタスクは追加されたか。追加された場合、いつ追加され、そのステータスの傾向は何か。
- 正常にクローズされたタスクの傾向は何か。イテレーションの最終期限に間に合わせることはできるか。

---

## 予定された最終期限に間に合わせるためのタスク管理

最終期限が近づいているが、「保留中」のタスクが残っていて、受け入れテストをまだ実行していない場合、ユーザーまたはマネージャーは以下について決定することができます。

- 重要度の低いタスクの中でどれを切り捨てることができるか。
- どんなタスクを今後のイテレーションに移動できるか。
- イテレーションの最終期限を延期できるか。
- このイテレーションを担当するチームメンバーをさらに増やす必要があるか。

予想されるイテレーションの終了日は次の場所に表示されます。

- [イテレーションの編集] ページ。137 ページの「イテレーションの編集」を参照。
- イテレーションのステータス レポート。174 ページの「[イテレーションのステータス] レポート」を参照。
- MS Project。MS Project の「プロジェクト ガイド」を参照。

タスクの移動については、137 ページの「イテレーションの編集」を参照してください。

---

## イテレーションの品質保証

特定のイテレーション中で作業されたすべての要求、欠陥、およびタスクには、自動テストまたは手動テストを作成し、プロジェクトの品質を保証するためにテストを実行する必要があります。イテレーションのステータス レポートに表示されるデータに基づいて次の情報を判断できます。

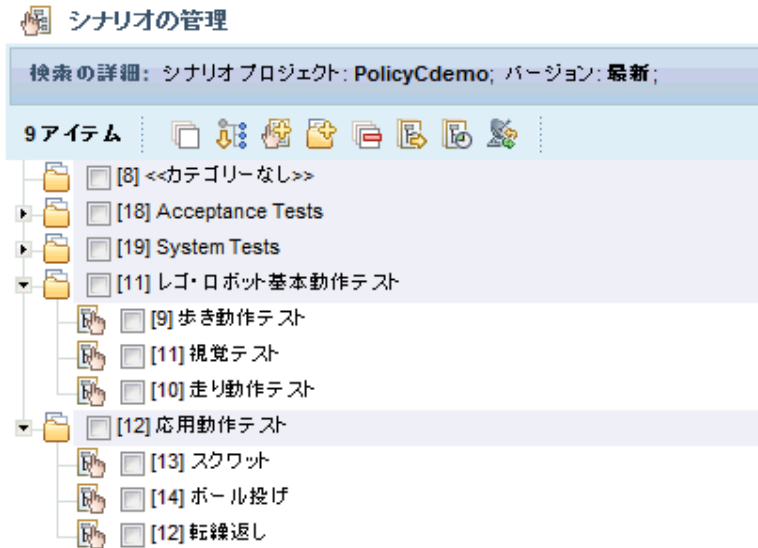
- プロジェクトまたはそのイテレーションに適切なテストが作成されたかどうか。
- 作成された場合、テストは実行されたか。テストは成功したか。

詳細については 174 ページの「[イテレーションのステータス] レポート」を参照してください。

## 受け入れテストの定義

Project Center には、手動テスト シナリオのリポジトリがあります。受け入れテストをこのリポジトリに格納するのは良いプラクティスです ( 図 13)。

図 13: シナリオの管理



受け入れテストの目標は、プロジェクトのすべての機能を徹底的にテストすることです。これには、プロジェクト中には存在するが、この特定のプロジェクト リリース用に開発されたのではない機能も含まれます ( 受け入れテストはこの点で「イテレーションの品質保証」と異なります。イテレーションの品質保証では、特定のイテレーションで開発された最近の機能だけをテストします )。

企業のポリシーに従って、プロジェクトの受け入れテストは次のタイミングで作成されます。

- 要求仕様の間に収集されて作成される ( 要求に割り当てられる )。
- 開発中に収集されて作成される。
- 開発が完了した後に収集されて作成される。

プロジェクト リリースの前に、マネージャーはプロジェクトのリリース候補にどの受け入れテストを実行するかを決定すべきです。

ユーザーあるいはマネージャーは、選択した受け入れテストを実施するテスト セッションをスケジューリングし、各テストを実行するオーナーを指名すべきです。ユーザーあるいはマネージャーは、テスト セッションの最中に進捗とステータスを頻繁にモニタリングすることができます。テスト セッションの結果に基づいて、マネージャーはプロジェクトの品質が十分かどうか、それともしらに改善が必要かどうかを判断すべきです。

## 欠陥の修正

古いバージョンやベータ版のアプリケーションをチーム メンバーまたは顧客がテストし、欠陥やフィーチャー リクエストをバグ トラッキング システムにレポートする場合があります。その場合、ユーザーまたはマネージャーは、特定の欠陥 / フィーチャー リクエストを Project Center データベースの既存の要求にリンクし、要求を開発者に割り当てて修正を依頼することができます。そうす

れば、すでに実装済みの要求で欠陥が発見された場合、その欠陥の修正を割り当てられた開発者は、要求に関連する履歴データを取得できます。取得できる履歴データには、ソースコード、自動テストと手動テストのシナリオ、およびこの要求のテスト実行のステータスなどが含まれます。そのため、開発チームの作業は格段に容易になり、作業スピードも向上します。

ユーザーまたはマネージャーは、要求ステータスレポートと欠陥/改善ステータスレポートを参照することで、どの要求に新しく欠陥が割り当てられたかを正確に把握することができます。また、修正がスケジュールリングされた欠陥や、すでに修正された欠陥も参照することができます。

---

## 結果のテスト

プロジェクトステータスレポートを参照すると、新しくレポートされた欠陥/フィーチャーリクエストとタスクのうち、完了したもの、テストされていないもの、および再テストが必要なものを確認できます。理想的には、すべてのプロジェクト成果物(要求、欠陥/改善、タスク)をテストする必要があります。

詳細については 173 ページの「プロジェクトステータスレポート」を参照してください。

# 開発プロセスの改善

通常、Development Testing Platform での作業の第三段階には次のステップが含まれます。

- 開発プロセスの最適化

## 開発プロセスの最適化

プロジェクトが完了して製品がリリースされたら、プロジェクトの開発履歴データを解析し、製品あるいは開発プロセスを改善するための修正が可能かどうかを判断することができます。



図 14: Project Center - 計画とトラッキング

Project Center では、[ レポート ] メニューから多彩なレポートにアクセスすることができます。レポートを解析することによって、以下の疑問に対する回答を得ることができます。

- どのタスクに最も時間がかかったか。
- それは特定の種類のタスクか（スタンドアロン、設計 / 計画、実装、要求の実装、欠陥 / 改善の修正、品質、要求レビューなど）。特定の種類である場合、今後の欠陥の数を減らすために、これまでよりもテストに力を入れるべきか。
- それは特定の要求の実装だったか。その場合、この要求によって得られた機能は、作業にかかった労力に見合っていたか。
- 予定された作業時間と実際の作業時間との乖離が最も大きかったタスクはどれか。
- それは特定のタイプのタスクか。
- チームメンバーの中に、タスクの最終期限に何度も大幅に遅れた者はいるか。
- 最も時間がかかったタスクはどれか。あるいは乖離が最も大きかったタスクはどれか。
- 今後、同様のプロジェクトをもっと短いイテレーションに分割するべきか。
- プロジェクトの後半の段階で欠陥が発見された要求は、初期の段階でテストされたのか。
- テスト（自動と手動の両方）は適切に作成されたか。
- イテレーションが開始した後に、新しいタスクが追加されたケースはあるか。ある場合、タスクはいつ追加されたのか。追加されたタスクにはどういった傾向があるか。
- 問題なくクローズされたタスクにはどういった傾向があるか。
- イテレーションの最終期限に間に合わせることはできるか。

以下のレポートによってこれらの疑問に回答することができます。

レポート	説明
[ 要求のコード レビュー ] レポート	要求とタスクの観点から、コード レビュー プロセスの概要および詳細情報を表示します。
[ 要求のステータス ] レポート	特定の要求について、実装の進捗とテスト ステータスを表示します。
[ 要求テスト マトリックス ] レポート	要求および関連するテストを表示します。
[ 欠陥 / 改善の統計 ] レポート	異なる観点 ( コンポーネント、クローズ バージョン、存続期間、オーナー ) から欠陥と改善を表示します。
[ 欠陥 / 改善のステータス ] レポート	選択したプロジェクトについてレポートされた欠陥と改善を表示します。
イテレーションレポート	表示されているプロジェクトのイテレーションとタスクについて、概要および詳細情報を表示します。
[ タスク アクティビティ ] レポート	特定の期間中に作業されたプロジェクト タスクの一覧を表示します。
[ タスクの統計 ] レポート	タスクの実装のステータスについて全体像を表示します ( タイプごと、オーナーごと、上位 10 の欠陥、改善と要求、予定作業時間と実作業時間のヒストグラム )。
[ テストの概要 ] レポート	プロジェクト中のテストおよびプロジェクト成果物 ( 要求、欠陥、タスク ) との関係を表示します。
[ テスト セッション ] レポート	選択したプロジェクトで実行されたテスト シナリオを表示します。

# Project Center の基本操作

このセクションの内容：

- GUI の詳細
- Project Center での検索操作
- CSV へのデータのエクスポート
- タスクのタグ付け
- 自動テストと Project Center との関連付け
- Project Center 成果物（欠陥、要求、タスク、テスト）の関連付け
- ストーリー ポイント
- 電子署名 / 変更理由の入力とトラッキング

---

## GUI の詳細

Project Center のホーム ページは次のセクションから構成されます。

- トップ パネル
- ツールバー

---

### トップ パネル

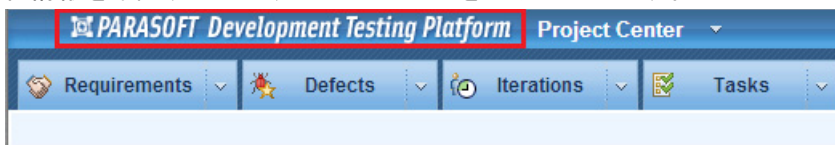
Parasoft Development Testing Platform は以下のアプリケーションから構成されます。

- Report Center
- Project Center
- License Server
- Team Server
- User Administration

---

### ライセンスの詳細の確認

ライセンスの詳細情報を確認するには、Parasoft ロゴをクリックします。



Control Center のリンクが表示されます。

[\[License configuration\]](#)

[\[DNS Services configuration\]](#)

Concerto 4.2 build (20101216 16:39:11 (Concerto\_4\_3\_branch))



[License Configuration] リンクをクリックすると、ライセンスについての詳細情報が表示されます。現行のライセンスを所有していることを確認してください。

#### Development Testing Platform License

License status: [License OK](#).

Development Testing Platform version: DTP 5.0  
Machine Id: LINUX2-741a75b2

#### License details:

Expiration date: unlimited

Password:

Features:

- DTP
- DTP Report Center
- DTP Project Center
- DTP Policy Center
- DTP Team Server
- DTP License Server
- DTP License Server additional 8 in subnet
- DTP License Server unlimited in subnet

Max tokens: 1

[\[Configure license\]](#)

## ヘルプ

[ ヘルプ ] ボタンをクリックすると、HTML 形式の『Development Testing Platform ユーザー マニュアル』が表示されます。

## 管理

[ 管理 ] リンクをクリックすると、管理者用ページが表示されます。



管理者用ページでは Project Center の重要な設定を行うことができます。

図 15: 管理者用ページ



次の操作が可能です。

- プロジェクトの検索。
- プロジェクトの追加。詳細については 230 ページの「プロジェクトの作成と構成」を参照。
- Eメールによるタスク通知の設定。詳細については 272 ページの「タスクと欠陥の通知メールの設定」を参照。
- 成果物の名前の変更。詳細については 274 ページの「成果物の名前変更」を参照。

**注意！** プロジェクトを検索するには、完全なプロジェクト名またはプロジェクト名の一部を入力します。[プロジェクト名] フィールドが空の状態を検索すると、既存のすべてのプロジェクトが検索結果に表示されます。

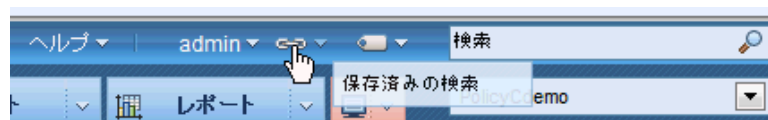
## ログアウト

[ ログアウト ] ボタンをクリックすると、Development Testing Platform からすぐにログアウトすることができます。

## クイックリンク

[ クイックリンク ] ボタンをクリックすると、保存した検索を参照することができます。

図 16: [クイックリンク] ボタン



保存した検索の詳細については 51 ページの「Project Center での検索操作」を参照してください。

## タグ

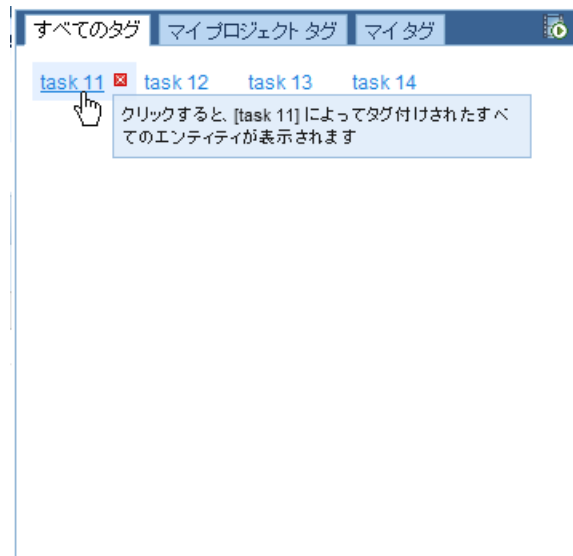
[ タグ ] ボタンをクリックすると、定義したすべてのタグを参照できます。

図 17: [ タグ ] ボタン



詳細については 56 ページの「タスクのタグ付け」を参照してください。

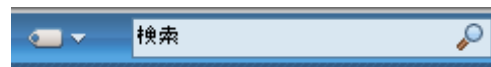
図 18: タグ ページ



## 検索

[ 検索 ] フィールドを使用すると、プロジェクト成果物（要求、欠陥 / 改善、イテレーション、タスク、およびテスト）をすばやく探すことができます。

図 19: [ 検索 ] フィールド



検索したいプロジェクト成果物の ID か、または検索したい任意の文字列を入力することができます。

検索結果は、検索文字列のヒット数でソートされます。

## ツールバー

ツールバーには以下のものがあります。

- ドロップダウン メニューがあるタブ。要求、欠陥、イテレーション、テストなどの成果物に簡単にアクセスすることができます。

- [マイ タスク] タブ。ユーザーに割り当てられたタスクが表示されます。
- デフォルト プロジェクトのコンボ ボックス。自分のプロジェクトを指定してください。Project Center のページを参照するときにデフォルトとして使用されます。

注意：ツールバーとステータス バーは、Project Center のすべてのページで共通です。

## 要求

[要求] タブをクリックすると、作業中のプロジェクトにある要求のリストが表示されます。

図 20: 要求のリスト

ID	名前	説明	ランク	ステータス
93	Easy user registration in the shop	It should be comfortable and almost seamless for th...	50000 - M	作業中
25	本体の組み上げ		50000 - M	作業中
92	Extra security needed	We should use all state-of-the-art technologies to ...	50000 - M	作業中
95	Browsing through the shop products	It should be easy and intuitive for customers to br...	50000 - M	作業中
27	音声認識と視覚性能		50000 - M	承認待ち

[要求] タブのドロップダウン メニューから、追加、インポート、管理といった要求に対する操作を行うことができます。次の操作が可能です。

- **新規**：要求を新規作成します。
- **検索**：既存の要求を検索します。
- **管理**：要求を管理します。
- **インポート**：既存の要求をインポートします。

## 欠陥

[欠陥] タブをクリックすると、作業中のプロジェクトにある欠陥のリストが表示されます。

図 21: 欠陥



[欠陥] タブのドロップダウン メニューの [検索] をクリックすると、欠陥を検索できます。

---

## イテレーション

[イテレーション] タブをクリックすると、作業中のプロジェクトにあるイテレーションのリストが表示されます。

[イテレーション] タブのドロップダウンメニューから、イテレーションの計画、新規作成、既存のイテレーションの検索、およびインポートを実行することができます。

---

## タスク

[タスク] タブをクリックすると、作業中のプロジェクトにあるタスクのリストが表示されます。

[タスク] タブのドロップダウンメニューから、タスクの新規作成、既存のタスクの検索、タスクの管理、およびインポートを実行することができます。

---

## テスト

[テスト] タブをクリックすると、作業中のプロジェクトにあるテストのリストが表示されます。

[テスト] タブのドロップダウンメニューから、テストの新規作成、既存のテストの検索、およびテストの管理を行うことができます。

---

## レポート

[レポート] タブをクリックすると、作業中のプロジェクトにある要求のレポートが表示されます。

[レポート] タブのドロップダウンメニューから、プロジェクトのステータス、タスク アクティビティ、および テスト セッションを参照することができます。

Project Center のレポートの詳細については 172 ページの「Project Center のレポート」を参照してください。

## ワークスペース / マイ タスク

[ ワークスペース / マイタスク ] タブでは、自分に割り当てられたタスクにすばやくアクセスすることができます。

図 22: ワークスペース : マイタスク

ワークスペース: 個人のタスク

作業中 ?

2アイテム

ID	名前	ステータス	要テスト	テスト	優先度	反復	S.
113	User profile page - GUI front-end	作業中	true	未テスト	中	29	
97	ConcertoとXtestの連携の調査	作業中	true	未テスト	中		

最近の5タスク ?

5アイテム

ID	名前	ステータス	テスト	優先度
160	今井がSOAtestから登録したタスク 8. 27	完了	未テスト	中
602	要件のレビュー: [115] R1	オープン	不要	中
601	要件のレビュー: [115] R1	オープン	不要	中
600	要件のレビュー: [116] R2	オープン	不要	中
599	要件のレビュー: [116] R2	オープン	不要	中

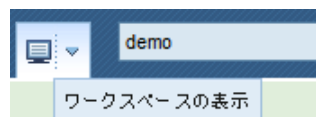
予定 ?

20アイテム

ID	名前	ステータス	テスト	優先度
117	Coordination and planning activities	オープン	不要	中
119	Implement 要件 113: Support for AmEx	オープン	未テスト	中

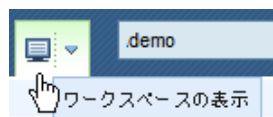
[ ワークスペース / マイタスク ] タブは、作業中のタスクの数によって違う色で表示されます。作業中のタスクがない場合、青で表示されます。

図 23: 青



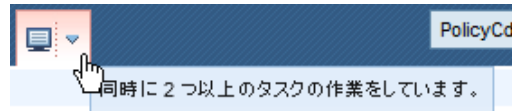
作業中のタスクが 1 個の場合、緑で表示されます。

図 24: 緑



作業中のタスクが 2 個以上ある場合、赤で表示されます。

図 25: 赤



このタブをクリックして、以下の 2 つのページにアクセスすることができます。

1. [ワークスペース：マイ タスク] ページ
2. マイ タスク クイック ビュー

### [ワークスペース：マイ タスク] ページ

モニターのアイコン (図 26) をクリックすると、自分の [ワークスペース：マイ タスク] ページに移動します。

図 26: モニター



ステータス (作業中、予定済み、最近 / 完了) ごとに自分のタスクを参照できます。以下の方法で [マイ タスク] ページから自分のタスクのステータスを変更できます。

- 表の右端にある十字の矢印のアイコンをクリックし、適切なステータスにタスクをドラッグアンドドロップします。たとえば、「予定」から「作業中」にタスクを移動します。

図 27: 十字の矢印のアイコン



- 表の左端の鉛筆のアイコンをクリックし、[ステータス] ドロップダウンメニューからステータスを選択します。

図 28: 鉛筆のアイコン



- 更新するタスクをダブルクリックします。タスクがハイライト表示されます (図 29)。
- [ステータス] カラムに移動し、ドロップダウンメニューからステータスを選択します。ステータスを変更したら、左側の緑のチェックマークをクリックして変更をコミットします。

図 29: ステータスの変更

	ID	名前	ステータス	テスト
✓ ✕	<a href="#">113</a>	User profile page - GUI front-end	作業中 ▼	true ▼
✎	<a href="#">97</a>	ConcertoとXtestの連携の調査	オープン <b>作業中</b> 完了 キャンセル	true

## マイ タスク クイック ビュー

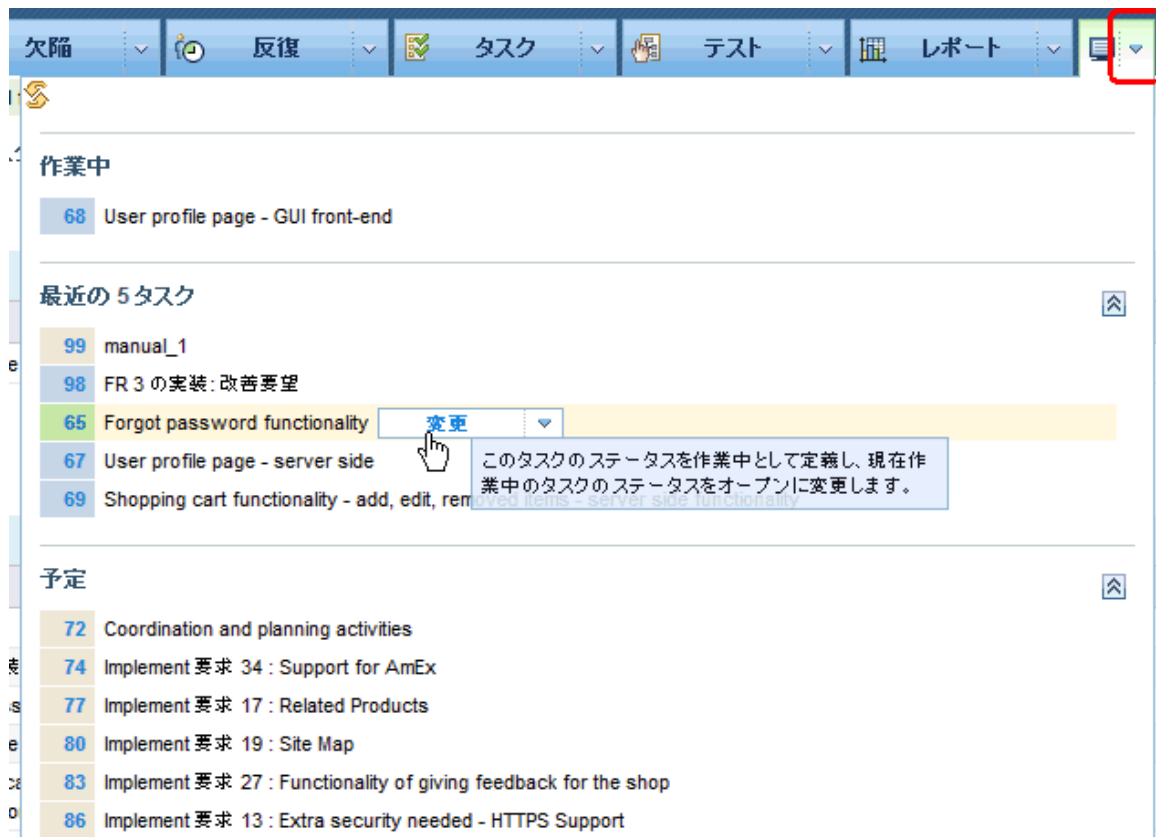
モニター アイコンの矢印をクリックすると、マイ タスク クイック ビューが表示されます。どのページからでも自分が作業しているすべてのタスクをすぐに参照できます。

タスクの短いリストが表示されるはずですが ( 図 30 を参照 )。

タスクの上にマウス ポインターを置くと、[ 変更 ] ボタンが表示され、そのタスクのステータスを変更できます。

**注意：** [ 変更 ] ボタンをクリックすると、現在のタスクのステータスがすぐに変更されます。「予定」のタスクは「作業中」に変わります。「作業中」のタスクは「オープン」に変わります。

図 30: マイ タスク クイック ビュー





## ヒント


[ ヒント ]  をクリックすると、ポップアップ ウィンドウが開き、現在のページについての重要なヒントが表示されます。このポップアップ ウィンドウは通常のヒントのページに似ています。

図 31: ヒント



## Project Center での検索操作

Project Center では、タスク、要求、欠陥など、さまざまな種類の成果物を扱います。成果物を探すために使用できる検索条件は、成果物の種類によって異なります。ただし、条件の変更や検索の保存など、基本的な機能はすべての検索で同じです。

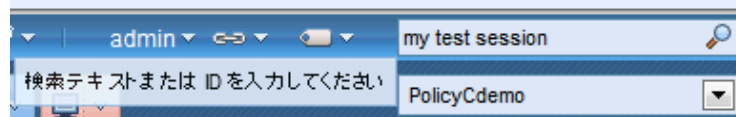
Project Center での検索には次のアクティビティが関係します。

- Project Center 全体に対するクイック検索
- 特定タイプの成果物の検索条件
- 検索条件の保存（オプション）
- 保存した検索のロードと削除

### Project Center 全体に対するクイック検索

Project Center の各ページの右上にある検索フィールドでは、Development Testing Platform の成果物 ID または文字列を検索することができます。

- 要求、欠陥、イテレーション、タスク、シナリオ、およびテスト セッションの ID
- 要求、欠陥、イテレーション、タスク、シナリオ、およびテスト セッションの名前と説明
- 要求、欠陥、およびタスク中のコメント
- シナリオのステップの説明




### 特定タイプの成果物の検索条件

デフォルトの検索条件を変更し、Project Center で他のアイテムを参照するには、次の操作を行います。

1. [ 要求 ]、[ 欠陥 ]、[ タスク ]、または [ テスト ] タブのいずれかをクリックして、その成果物のページに移動します。



2. [ 検索 ] ボタン  をクリックします。

選択した成果物の検索ページが表示されます。

**注意：** デフォルトでは、その成果物を表示するページでデータを投入するために使用される検索条件が表示されます。

3. 次のいずれかの操作を行って検索条件を変更します。

- 同じような検索条件を定義するには、必要な変更を行います。
  - まったく新しい検索条件を定義するには、[ 消去 ] をクリックしてから必要な変更を行います。
4. [ 検索 ] をクリックします。

検索ダイアログが閉じ、新しい条件を使って検索された成果物が表示されます。

**注意：** 検索条件は、ステータス バーの左側にも表示されます。

## 検索条件の保存 ( オプション )

検索の保存機能を使用すると、普段よく扱うプロジェクト成果物 ( 要求、欠陥、イテレーション、タスク、テスト シナリオ ) に簡単にアクセスすることができます。たとえばチーム リーダーであれば、特定のプロジェクトにおける作業中のタスクをすべて検索することで、プロジェクトの状況を確認することができます。他によく使用される検索として以下があります。

- 現在のイテレーションでオープンなタスクを検索する
- 完了しているがまだテストされていないタスクを検索する
- いくつかのプロジェクトにまたがって複数のイテレーションを検索する
- まだ実装されていない要求をすべて検索する
- 特定のユーザーが定義したテスト シナリオをすべて検索する

検索条件を保存するには、次の操作を行います。

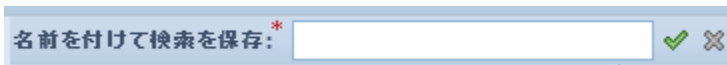
1. 保存する検索条件を使って検索を実行します。
2. [ 現在の検索を保存 ] ボタンをクリックします。

図 32: [ 現在の検索を保存 ] ボタン



[ 名前を付けて検索を保存 ] フィールドが表示されます。

図 33: [ 名前を付けて検索を保存 ] フィールド



3. たとえば「My Project Tasks」などのように、保存する検索条件の名前を [ 名前を付けて検索を保存 ] フィールドに入力します。
4. 次のいずれかの操作を行います。
  - [ 保存 ] ボタンをクリックして検索条件を保存します。
  - [ キャンセル ] ボタンをクリックして操作をキャンセルします。

## 保存した検索のロードと削除

保存した検索条件にはいつでもアクセスすることができます。また、必要に応じて削除することも可能です。

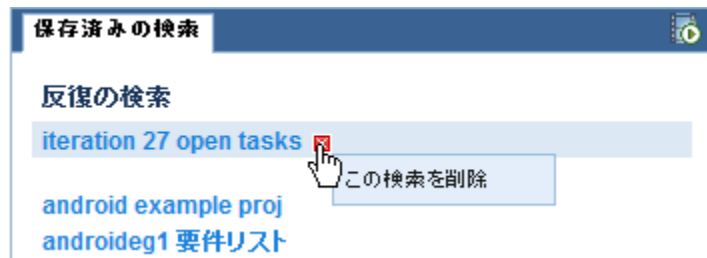
保存した検索をロードするには、[ 保存済みの検索 ] をクリックして検索を選択します。

図 34: 保存済み検索の選択



保存した検索を削除するには、次の操作を行います。

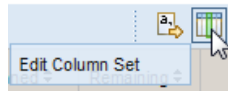
1. 削除する検索の上にマウス ポインターを置きます。
2. [ この検索を削除 ] ボタンをクリックします。



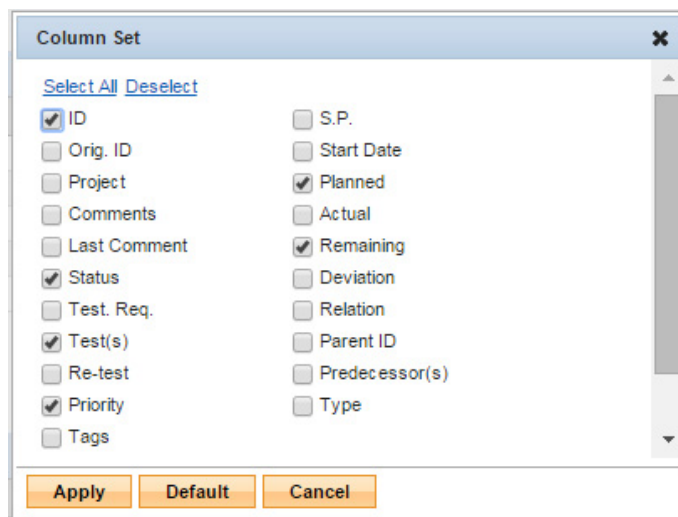
# CSV へのデータのエクスポート

表形式のデータの場合、データを CSV ファイルにエクスポートすることができます。次の操作を行います。

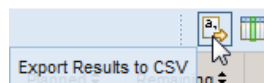
1. [ カラム セットの編集 ] ボタンをクリックして表示するカラムを確認し、保存したいデータが確実に含まれるようにします。



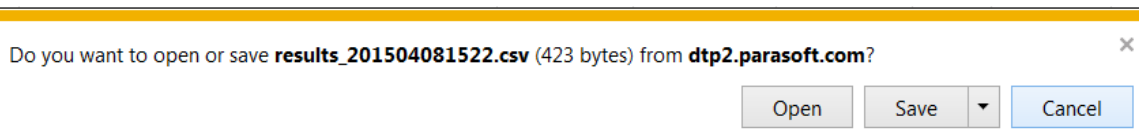
2. CSV ファイルに含めるカラムを選択し、[ 適用 ] ボタンをクリックします。



3. [ 結果を CSV にエクスポート ] ボタンをクリックします。



ブラウザと設定によっては、ファイルを開くか保存するかの選択を求められます。

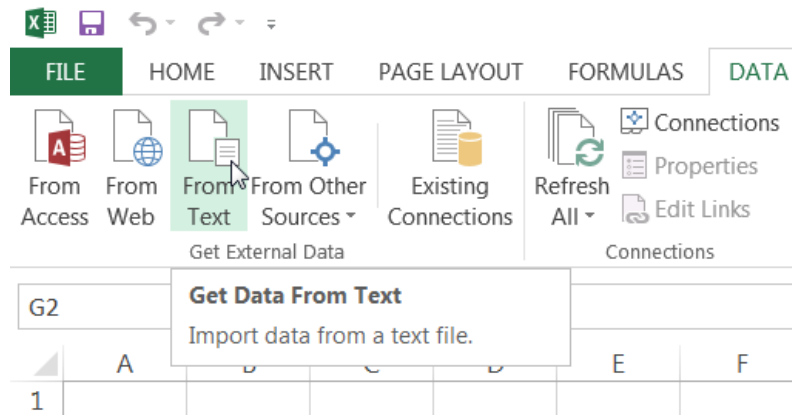


4. ファイルを保存します。保存したファイルは Microsoft Excel などの表計算アプリケーションにインポートできます。

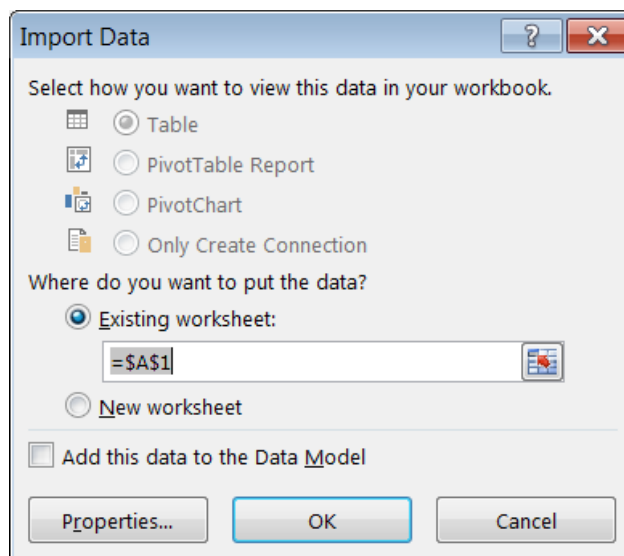
## Project Center からエクспортされた CSV ファイルを開く

ダウンロードした CSV ファイルをダブルクリックして表計算アプリケーションで開くことができます。ただし、たとえば非標準文字が含まれているなど、問題がある場合は、表計算アプリケーションで CSV ファイルを開いてからインポートする必要があります。

1. 空の Excel 文書を開きます。
2. [ データ ] タブをクリックし、[ 外部データの取り込み ] リボンで [ テキスト ファイル ] をクリックします。



3. ダウンロードした CSV ファイルを選択して [ インポート ] をクリックします。
4. ウィザードで [ カンマやタブなどの区切り文字によってフィールドごとに区切られたデータ ] を選択し、[ 次へ ] をクリックします。
5. [ 区切り文字 ] を [ カンマ ] に変更し、[ 完了 ] をクリックします。
6. データのインポート先を選択し、[ OK ] をクリックします。



意図したとおりにデータが表計算アプリケーションに表示されます。

## タスクのタグ付け

Project Center では、タグを使って各タスクを特徴づけることができます。タグはタスクを説明するキーワードであり、タグ属性として分類されます。この機能はソフトウェア開発ライフ サイクル (SDLC) を単純化し、プロジェクト全体に適切なワークフローを確保するのに役立ちます。

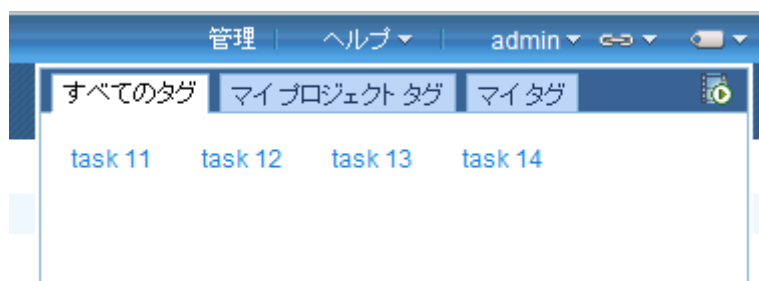
- タグ パネル
- タグによるタスクのマーキング

## タグ パネル



[ タグ ] ボタンは、Project Center のメイン ページの右上にあります。このボタンをクリックすると、Development Testing Platform で定義されたすべてのタグが表示されます。

図 35: タグ パネル



タグは 3 種類のカテゴリによってフィルタリングできます。

- **すべてのタグ**：作成されたすべてのタグが表示されます。
- **マイ プロジェクト タグ**：自分のプロジェクトで使用されるすべてのタグが表示されます。
- **マイ タグ**：自分が作成したタグだけが表示されます。

フォント サイズはそのタグの人気を表します。タグ名の横の括弧内の数字は、そのタグでマーキングされたタスクの数を示します。選択したタグ名をクリックすると、そのタグに関連付けられたすべてのタスクが表示されます。

## タグによるタスクのマーキング

[ タスクの編集 ] ページの [ 全般 ] タブで既存のタグ名のリストを見ることができます。コンボボックスにタグ名とその人気が表示されます。既存のタグを選択するか、新しいタグを作成できます。新

新しいタグを作成するには、固有のタグ名を [ タグの追加 ] フィールドに入力して [ タグの追加 ] リンクをクリックした後、タスクの [ 保存 ] ボタンをクリックしてタスク全体を保存します。

図 36: タグの追加

タグは、Project Center 中のどのタスクに対しても作成できます。

複数のタスクを特定の 1 つのタグで同時にマーキングすることができます。複数のタスクに対するタグを作成するには、[ タスク ] タブをクリックし、左側のチェックボックスをオンにして適切なタスクを選択します。そして [ 一括更新 ] をクリックします。

	ID	名前	ステータス
<input checked="" type="checkbox"/>	35	要件 25 の実装: 本体の組み上げ	完了
<input checked="" type="checkbox"/>	36	要件 27 の実装: 音声認識	作業中



[ タグ ] フィールドでタグ名を指定し、[ 保存 ] をクリックします。選択したすべてのタスクが特定のタグと関連付けられます。

図 37: タグの選択

更新のために選択されたタスク

ID	名前	プロジェクト	ステータス	テストが必要	優先度	要求	欠陥	反復	
35	要件 25 の実装: 本体の組み上げ	PolicyCdemo	完了	✓	高	25		8	naga
36	要件 27 の実装: 音声認識	PolicyCdemo	作業中	✓	中	27		8	kylie

関係: ---

全般

プロジェクト: --- 反復 ID:   反復なし

優先度: --- 監督者:  [表示] [閉去]

オーナー:  [表示] [閉去]

ステータス:  オープン  作業中  完了  キャンセル

タグ:

テストが必要: task 11 (x1)  
task 12 (x1)  
task 13 (x1)  
task 14 (x1)

コメント:

許可されるリンクの書式: http|ftp|https://company.com, www.company.com, @task NNN, @req NNN (@task 111, @REQ 222)

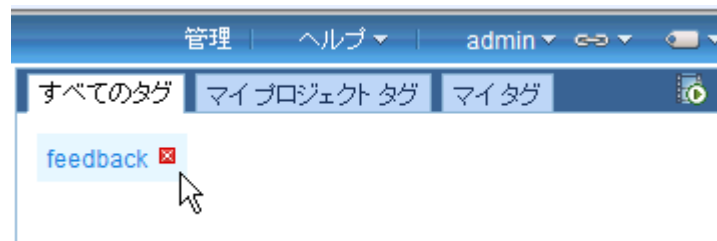
作業量

ストーリー ポイント:

タグの横の赤い × 印をクリックしてタグを削除できます。この赤い × 印は次の 2 箇所に表示されます。

- [ タグ ] パネル: タグ全体を削除すると共に、このタグに対するすべてのタスクの関連付けを削除します。
- [ タスクの編集 ] ページ: 選択したタグとそのタスクの間の関係だけを削除します。

図 38: タグの削除



タグはタスクのフィルタリングに役立ちます。[ タスク検索 ] ページでタグ フィールドの隣の [Not] チェックボックスをオンにすると、指定したタスクに関連しない、すべてのタスクが分かります。

図 39: [Not] チェックボックス

#### タスクの関係

The screenshot shows a search form for tasks. On the left, there are several search criteria: '反復 ID:' (Iteration ID) with a search button, '反復内にある:' (Within iteration) with a dropdown menu set to '任意' (Any), '要求 ID:' (Requirement ID) with a search button, '欠陥 ID:' (Defect ID) with a search button, and 'タグ:' (Tag) with the value 'task 11,' and a checked checkbox labeled 'Not'. On the right, there is a 'タスク プロジェクト:' (Task Project) dropdown menu with options: 'PolicyCdemo', 'ReportCenterTest\_Project', 'Shapes', 'SOAtestプロジェクト', and 'Software Process'. Below it is a 'タスク オーナー:' (Task Owner) field with a search button and a '作成者:' (Creator) field. A tooltip points to the 'Not' checkbox, stating '指定の条件に一致しない項目だけを表示する' (Display only items that do not match the specified conditions).

タグは任意の単語で記述でき、任意のタスク属性を表現できます。

---

# 自動テストと Project Center との関連付け

開発者が作業するすべてのプロジェクト成果物（要求、欠陥 / 改善、タスク）では、自動テストと手動テストを用意してプロジェクトの終了前にテストを実行する必要があります。このテストによって、最終的な製品の品質と信頼性が保証されます。

要求ステータス、欠陥 / 改善ステータス、イテレーション ステータス、欠陥 / 改善の検索、タスクの検索、イテレーションの編集といったさまざまな Project Center ページから、プロジェクト成果物のテスト ステータスを参照できます。

要求、欠陥 / 改善、またはタスクと自動テストとを関連付けるには、次の操作を行います。

- 適切な要求、欠陥 / 改善、タスク ID と自動テスト ケースとを関連付けます。次を参照してください。
  - 60 ページの「要求とテスト ケースとの関連付け」
  - 63 ページの「欠陥 / 改善とテスト ケースとの関連付け」
  - 66 ページの「タスク ID とテスト ケースとの関連付け」
- 適切な Parasoft 製品を使ってテストを実行します。
  - Jtest: Java コードをテストします。
  - C++test: C/C++ コードをテストします。
  - dotTEST: .NET コードをテストします。
  - SOAtest: SOA、Web、および クラウド アプリケーションをテストします。

**重要！** テストを実行する前に、Development Testing Platform サーバーにテスト結果を送信するよう、適切な Parasoft 製品を構成する必要があります。Development Testing Platform サーバーのホスト、ポート、およびプロジェクトの属性を定義して Parasoft 製品から Project Center（および Report Center）にテスト結果を送信する方法については『Development Testing Platform 管理者マニュアル』を参照してください。

特定の製品成果物に関連付けられた自動テストの実行結果を参照できます。詳細については 204 ページの「自動テストの詳細」を参照してください。

要求、欠陥 / 改善、またはタスクと自動テストとの関連付けは解除することもできます。68 ページの「テスト ケースと成果物の関連付けの解除」を参照してください。

---

## 要求とテスト ケースとの関連付け

たとえば、ID が 123 の要求に対して作業していて、その機能をテストするための自動テストを作成しているものとします。以下の Parasoft 製品を使ってこの要求とテスト ケースとを関連付けることができます。

- Jtest

- C++test
- dotTEST
- SOAtest

---

## Jtest

Jtest で要求とテスト ケースを関連付けるには、各 Junit メソッド ( テスト ケース ) に Javadoc コメント `@req 123` を追加します。

```
/**
 * @req 123
 */
@Test
public void myUnitTest()
{
    //test body
}
```

---

## C++test

C++test でテスト ケースを要求と関連付けるには、単体テストの各テスト メソッド名に接尾辞 `_REQ123` を付けます。

例 :

```
public void myUnitTest_REQ123()
{
    //test body
}
```

別のやり方として、C++test 単体テスト ケースの各テスト メソッドにコメントとして `@REQ 123` を追加してテスト ケースと要求を関連付ける方法もあります。

たとえば、123 および 487 とテストを関連付けるには、次のようにコメントを追加します。

```
/* CPPTEST_TEST_CASE_BEGIN test_case */
//@REQ 123
//@REQ 487
void TestSuite::test_case()
{
    //test body
}
/* CPPTEST_TEST_CASE_END test_case */
```

## dotTEST

dotTEST でテスト ケースと要求を関連付けるには、単体テストの各テスト メソッドに TestCaseREQ 属性を追加します。

```
[TestCaseREQ("123")]
    [Test]
    public void MyUnitTest()
    {
        //test body
    }
```

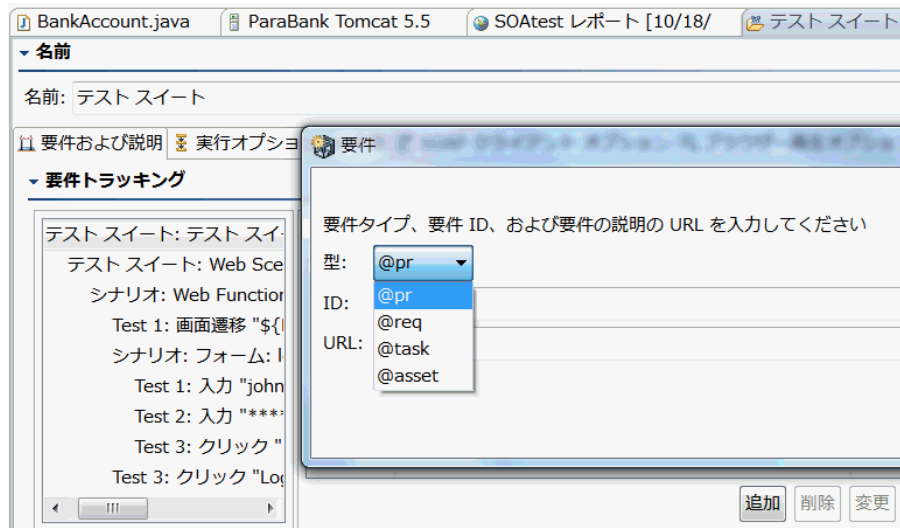
## SOAtest

SOAtest でテスト ケースを要求と関連付けるには、次の操作を行います。

1. [要件および説明] タブをクリックします。
2. 新しい要求にテスト ケースを割り当てるために [追加] ボタンをクリックします。

[要件] ダイアログが表示されます (図 40)。

図 40: [Requirement] ダイアログ



3. [型] ドロップダウン リストから @req を選択します。
4. [ID] フィールドに、たとえば「123」のように要求 ID を入力します。

**注意:** [URL] フィールドは空白のままにすることができます。

5. [OK] をクリックします。

SOAtest を使ってテスト ケースを実行した後、結果が Project Center に送信され、指定の要求と関連付けられます。

---

## 欠陥 / 改善とテスト ケースとの関連付け

開発者によって解決された欠陥と改善は、すべてテストされるべきです。たとえば、ID 456 の欠陥と ID 789 の改善に対して自動テストを作成しているものとします。以下の Parasoft 製品でテスト ケースを欠陥 / 改善と関連付けることができます。

- Jtest
- C++test
- dotTEST
- SOAtest

---

### Jtest

Jtest でテスト ケースを欠陥 / 改善と関連付けるには、各 Junit メソッド ( テスト ケース ) に Javadoc コメントの @pr 456 または @fr 789 を追加します。

例 :

```
/**
 * @pr 456
 */
@Test
public void myUnitTest()
{
    //test body
}
```

or

```
/**
 * @fr 789
 */
@Test
public void myUnitTest()
{
    //test body
}
```

## C++test

C++test でテスト ケースを欠陥 / 改善と関連付けるには、単体テスト ケースの各テスト メソッドの名前に次の接尾辞を追加します。

- `_PR456` (欠陥の場合)
- `_FR789` (改善の場合)

例：

```
public void myUnitTest_PR456()
{
    //test body
}
```

or

```
public void myUnitTest_FR789()
{
    //test body
}
```

別のやり方として、単体テスト ケースの各テスト メソッドにコメントとして次の属性を追加する方法もあります。

- `@PR 456` (欠陥の場合)
- `@FR 789` (改善の場合)

たとえば、欠陥 456 および 620 とテストを関連付けるには、次のように指定します。

```
/* CPPTEST_TEST_CASE_BEGIN test_case */
//@PR 456
//@PR 620
void TestSuite::test()case()
{
    //test body
}
/* CPPTEST_TEST_CASE_END test_case */
```

また、たとえば 改善 789 および 753 とテストを関連付けるには、次のように指定します。

```
//@FR 789
//@FR 753
void TestSuite::test_case()
```

```
{  
    //test body  
}
```

---

## dotTEST

dotTEST でテスト ケースを欠陥 / 改善と関連付けるには、単体テストの各テスト メソッドで次の属性を使用します。

- TestCasePR (欠陥の場合)
- TestCaseFR (改善の場合)

例：

```
[TestCasePR("456")]  
    [Test]  
    public void MyUnitTest()  
    {  
        //test body  
    }  
}
```

または

```
[TestCaseFR("789")]  
    [Test]  
    public void MyUnitTest()  
    {  
        //test body  
    }  
}
```

---

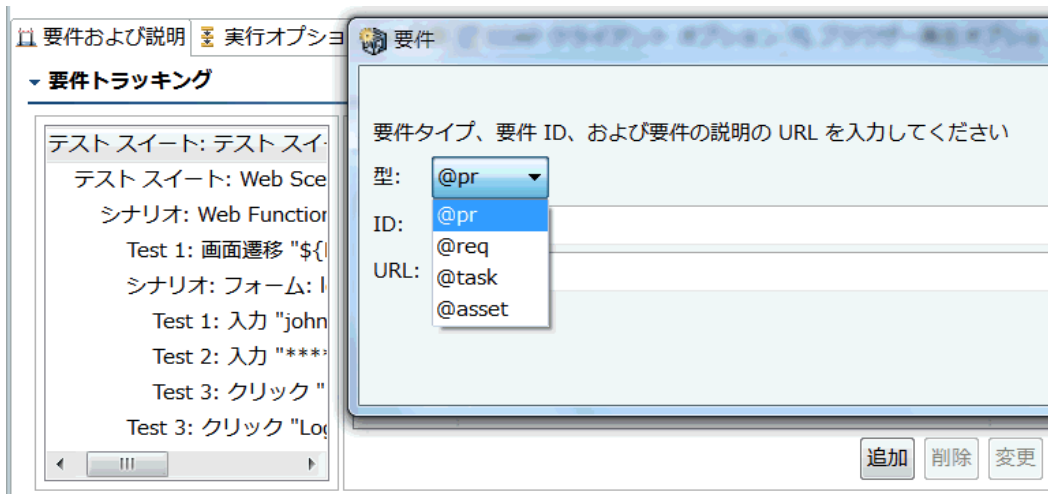
## SOAtest

SOAtest でテスト ケースを欠陥 / 改善と関連付けるには、次の操作を行います。

1. [要件および説明] タブをクリックします。
2. 新しい欠陥 / 改善にテスト ケースを割り当てるために [追加] ボタンをクリックします。  
[要件] ダイアログが表示されます (図 40)。



図 41: [要件] ダイアログ



3. [型] ドロップダウン リストから @pr (欠陥の場合) または @fr (改善の場合) を選択します。
4. 例に合わせて、[ID] フィールドに「456」または「789」を入力します。  
**注意:** [URL] フィールドは空白のままにすることができます。
5. [OK] をクリックします。

SOAtest を使ってテスト ケースを実行した後、結果が Project Center に送信され、指定の欠陥または改善と関連付けられます。

## タスク ID とテスト ケースとの関連付け

たとえば、ID 321 のタスクについて作業していて、このタスクに対する自動テストを作成しているものとして、次の Parasoft 製品でテスト ケースとタスクを関連付けることができます。

- Jtest
- C++test
- dotTEST
- SOAtest

### Jtest

Jtest でタスク ID とテスト ケースを関連付けるには、各 Junit メソッド (テスト ケース) に Javadoc コメント @task 321 を追加します。

例 :

```
/**
 * @task 321
 */
```

```
@Test
public void myUnitTest()
{
    //test body
}
```

---

## C++test

C++test でタスク ID とテスト ケースを関連付けるには、単体テストの各テスト メソッド名に接尾辞 `_TASK321` を付けます。

例：

```
public void myUnitTest_TASK321()
{
    //test body
}
```

別のやり方として、C++test 単体テスト ケースの各テスト メソッドにコメントとして `@TASK 321` を追加する方法もあります。

たとえば、タスク 321 および 420 とテスト ケースを関連付けるには、次のようにコメントを追加します。

```
/* CPPTEST_TEST_CASE_BEGIN test_case */
//@TASK 321
//@TASK 420
void TestSuite::test_case()
{
    //testcase body
}
/* CPPTEST_TEST_CASE_END test_case */
```

---

## dotTEST

dotTEST でタスク ID とテスト ケースを関連付けるには、単体テストの各テスト メソッドに `TestCaseTASK` 属性を追加します。

例：

```
[TestCaseTASK("321")]
    [Test]
    public void MyUnitTest()
    {
        //test body
    }
```

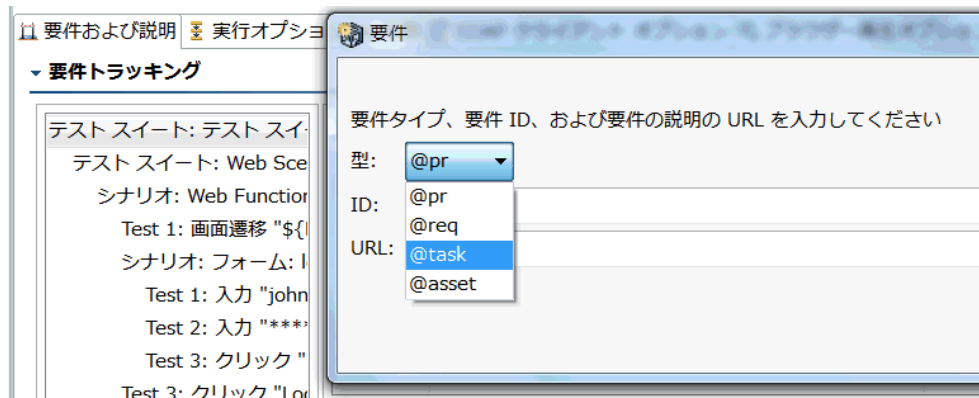
}

## SOAtest

SOAtest でタスク ID とテスト ケースを関連付けるには、次の操作を行います。

1. [ 要件および説明 ] タブをクリックします。
2. 新しいタスク ID にテスト ケースを割り当てるために [ 追加 ] ボタンをクリックします。  
[ 要件 ] ダイアログが表示されます ( 図 40)。

図 42: [ 要件 ] ダイアログ



3. [ 型 ] ドロップダウン リストから @task を選択します。
4. [ID] フィールドにタスク ID を入力します (たとえば「321」など)。  
**注意:** [URL] フィールドは空白のままにすることができます。
5. [OK] をクリックします。

SOAtest を使ってテスト ケースを実行した後、結果が Project Center に送信され、指定のタスク ID と関連付けられます。

## テスト ケースと成果物の関連付けの解除

テスト ケースを特定のプロジェクト成果物にもう関連付けるべきではないと判断した場合、[ 自動テスト ] タブの実行結果で関連付けを解除することができます。[ 自動テスト ] タブは、[ 要求の編集 ] ページ、[ 欠陥 / 改善の編集 ] ページ、および [ タスクの編集 ] ページに表示されます。

テスト ケースの関連付けを解除するには、テスト ケースに対応する [ このテストの割り当てを解除 ] ボタンをクリックします ( 図 43)。

図 43: [このテストの割り当てを解除] ボタン



テストが失敗した場合、関連付けを解除することは特に重要です。解除しなければ、このテストケースが関連付けられているすべての要求、欠陥 / 改善、またはタスクのステータスが「失敗」になります。

**重要！** この割り当ての解除は永続します。[このテストの割り当てを解除] をクリックすると、「このテスト ケースを、対応する要求、欠陥 / 改善、タスクに関連付けたくない」ということが Project Center に指示されます。適切な @reqID/prID/frID/taskID タグを使って次の自動テスト セッションでこのテスト ケースが実行される場合でも、この要求、欠陥 / 改善、タスクに関連付けられたテスト ケースとして表示されることはありません。

---

# Project Center 成果物 ( 欠陥、要求、タスク、テスト ) の関連付け

Project Center では、以下の操作によって相互に関連付けを指定したりトラッキングすることができます。

- ソース コードと成果物の関連付け
- 自動テストと要求または欠陥との関連付け
- 欠陥、改善、テスト、タスクの関連付け
- 欠陥とシナリオ実行の関連付け

---

## ソース コードと成果物の関連付け

Project Center は、特定のタスク、要求、または欠陥に対する作業中にソース コード リポジトリにコミットされたすべてのファイルとリビジョンをトラッキングすることができます。

要求とソース コードを関連付けるのは開発者だけです。開発者は次の点を考慮する必要があります。

- タスクに対して作業する場合、27 ページの「タスクに対する作業」の説明にある手順に従うべきです。

要求または欠陥にタスクがリンクされたら、タスクのソース コードが [ 要求 ] タブまたは [ 欠陥 ] タブにも表示されます。

- タスクではなく要求に対してだけ作業する場合、リポジトリにファイルをコミットするときにリビジョン コメントに次の文字列を含めるべきです ( "NNN" は要求 ID です )。

@req NNN

- 要求に関連する欠陥を修正する場合 ( そしてタスクに対して作業しない場合 )、71 ページの「欠陥、改善、テスト、タスクの関連付け」の説明にあるように、欠陥 / 改善を要求と関連付けるべきです。

欠陥 / 改善が要求とリンクされると、そのソース コードが要求にも表示されます。

[ コード ] タブには、以下のアイテムに対する作業中にソース管理システムにコミットされたすべてのファイル リビジョンが表示されます。

- 要求、それ自体。
- 要求に関連付けられている欠陥 / 改善。
- 要求に関連付けられているタスク。

---

## 自動テストと要求または欠陥との関連付け

ある特定の要求または欠陥に対して作業している間、Project Center はその要求 / 欠陥を実行するすべての自動テストをトラッキングします。Project Center で自動テストの結果を要求、欠陥 / 改善、またはタスクに関連付けるには、Parasoft 製品を使って何らかの関連付けを作成する必要があります。

ります。詳細については 60 ページの「自動テストと Project Center との関連付け」を参照してください。

## 欠陥、改善、テスト、タスクの関連付け

Development Testing Platform ではさまざまな成果物に対して相互に関連付けを指定することができます。たとえば：

- テスト シナリオを作成するときに、どの特定の要求または欠陥 / 改善をテスト シナリオで検証するかを指定できます。
- 要求が実装された後に欠陥 / 改善がレポートされることもあります。そのような欠陥 / 改善を、関連する要求に相互に関連付けることができます。

Development Testing Platform のさまざまなアイテムを相互に関連付けるには、次の操作を行います。

1. 相互に関連付ける成果物を開きます。たとえば、要求を別の成果物と相互に関連付ける場合、その要求のページを開きます。
2. この成果物が相互に関連するアイテム タイプのタブを開きます。たとえばテスト シナリオに関連付けるのであれば、[ シナリオ ] タブを開きます。
3. [ 割り当て ] ボタンをクリックします。次の画面が表示されます。

検索の詳細: シナリオプロジェクト: FDAdemo; バージョン: 最新; 保存済み検索: - デフォルト -									
10 アイテム <span style="float: right;">割り当て</span>									
<input type="checkbox"/>	ID	名前	バージョン	ステータス	作成者	シナリオ実行結果	実行者	テスト セッション ID	テストモ
<input type="checkbox"/>	3	LCD表示中にボタンをオンするとシャットダウンする	0	Ready for QA	admin	成功	naga	1	機能テスト
<input type="checkbox"/>	4	測定精度の確認	0	Ready for QA	admin	成功	naga	1	機能テスト
<input type="checkbox"/>	1	ボタンオン後2秒以内に初期状態を確認する	0	Ready for QA	admin	未実行		1	機能テスト
<input type="checkbox"/>	2	アイドル状態からシャットダウンまでの時間を確認	0	Ready for QA	admin	成功	admin	1	機能テスト
<input type="checkbox"/>	7	アイドル状態からシャットダウンまでの時間を確認 - Cloned	0	Ready for QA	naga	成功	masu	2	システムテスト
<input type="checkbox"/>	8	LCD表示中にボタンをオンするとシャットダウンする - Cloned	0	Ready for QA	naga	成功	masu	2	システムテスト
<input type="checkbox"/>	5	体温を測定15秒後に計算し、アラームを鳴らし、表示する	0	Ready for QA	admin	成功	naga	1	機能テスト

### 検索条件

場合によって、[ 検索条件 ] ページが開くことがあります。その場合、関連付ける成果物の条件を入力して [ 検索 ] をクリックします。

4. 適切な成果物の隣のチェックボックスを切り替えて、現在の成果物に割り当てます。
5. [ 割り当て ] をクリックします。

**相互に関連付ける成果物を新規に作成するには**

[ 割り当て ] をクリックして上記のステップを実行する代わりに、[ 新規 ] をクリックして目的の成果物 ( タスクまたはシナリオ ) を指定します。

## 欠陥とシナリオ実行の関連付け

シナリオを実行すると、実行を欠陥に割り当てたり、割り当てを解除するためのオプションが表示されます。

1. シナリオを実行します。テストの実行に関する詳細については 163 ページの「手動テストの実行」を参照してください。
2. ダイアログが更新されたら、[ 欠陥 ] タブをクリックします。
3. テストの実行に新しい欠陥を割り当てる場合は [ 新規 ] をクリックし、既存の欠陥に関連付ける場合は [ 割り当て ] をクリックします。

The screenshot shows the 'Defects' tab interface. At the top, there are tabs for 'Execution', 'Scenario Attachments', 'Defects', and 'Modification History'. Below the tabs, there are buttons for 'New' and 'Assign'. A table header is visible with columns for 'ID' and 'Name'. Below the table, it says 'NO DATA TO DISPLAY'.

シナリオ実行に関連付けられた欠陥は、欠陥の [ シナリオ実行 ] タブで割り当てを解除することができます。

The screenshot shows the 'Scenario Runs' tab interface. At the top, there are tabs for 'Requirements', 'Defects', 'Iterations', 'Tasks', 'Tests', and 'Reports'. Below the tabs, there are buttons for 'New', 'Search', and 'Import'. The main area shows 'Edit Defect/Enhancement: [0] a0'. Below that, there are tabs for 'General', 'Tasks', 'Requirements', 'Code', 'Automated Tests', 'Scenarios', and 'Scenario Runs'. A yellow warning message states: 'Only the Scenario Runs that are directly associated to this Defect/Enhancement are listed below.' Below the message, there are buttons for 'Unassign'. A table header is visible with columns for 'Scenario Run ID', 'Scenario Run Name', 'Scenario Version', 'Scenario Run Status', 'Scenario Run Result', 'Performed By', and 'Test Session ID'.

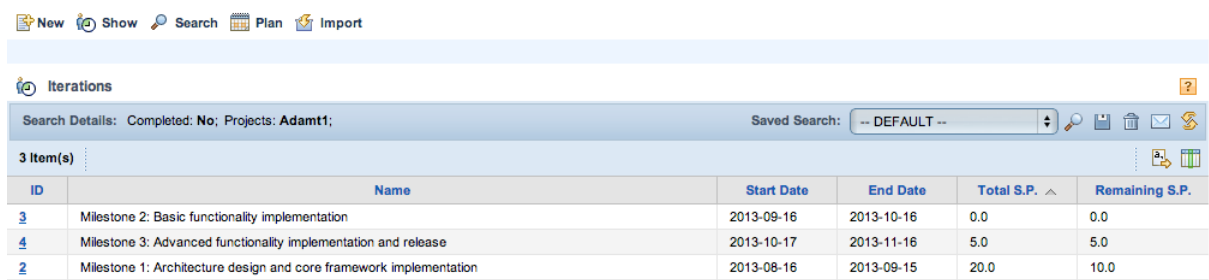
# ストーリー ポイント

Project Center では、要求、欠陥、タスクを含め、プロジェクト バックログ アイテムのためにストーリー ポイントを定義することができます。ストーリー ポイントは以下のページで入力および編集できます。

- 要求 / 欠陥 / タスクの編集
- イテレーションの計画 (" クイック変更 " オプションによって)
- 一括更新 (タスク)
- タスクの検索結果 ページ (" クイック変更 " オプションによって)

イテレーションの計画中にストーリー ポイントを使用することができます。たとえば、要求または欠陥 / 改善のタスクをスケジューリングするときに、アイテムのストーリー ポイントは特定のタスクの「予定時間」に変換されます。詳細については 114 ページの「イテレーションの計画」を参照してください。

各イテレーションに関連付けられたストーリー ポイントの完了状況を確認するには、Project Center メイン メニューの [ イテレーション ] タブをクリックします。



ID	Name	Start Date	End Date	Total S.P. ^	Remaining S.P.
3	Milestone 2: Basic functionality implementation	2013-09-16	2013-10-16	0.0	0.0
4	Milestone 3: Advanced functionality implementation and release	2013-10-17	2013-11-16	5.0	5.0
2	Milestone 1: Architecture design and core framework implementation	2013-08-16	2013-09-15	20.0	10.0

- **合計 S.P.:** イテレーションに含まれるすべてのバックログ アイテムに割り当てられたストーリー ポイントの合計です。
- **残 S.P.:** まだ完了していないバックログ アイテムに割り当てられたストーリー ポイントの値です。

## ストーリー ポイントの背景

ストーリー ポイントとは、機能を実装するのに必要な作業量を見積るために使用される単位です。この概念はアジャイル / スクラムに由来します。アジャイル / スクラムの世界では、ストーリー ポイントを使って、ユーザー ストーリーを実装するのに必要な作業量を見積ります (ユーザー ストーリーは要求に相当するものです)。

ストーリー ポイントは、ユーザー ストーリーの実装の難しさを示す数字として表されます (難しさの判断には、複雑さと作業量が考慮されます)。ある機能のストーリー ポイントが別の機能のストーリー ポイントの 2 倍である場合、その機能のユーザー ストーリーを実装するには 2 倍の作業量が必要です。

バックログ アイテムに使用する場合、ストーリー ポイントはそのバックログ アイテムを完了するために必要な作業時間を表しません。アイテムを完了するための予測作業時間は、機能の難しさだけでなく、担当者の能力にも依存します。



ストーリー ポイントを使ってプロジェクト バックログ アイテムの見積りを開始するには、チームメンバー全員が基準として参考にできるように、ベースラインとなるストーリーを探してそのストーリー ポイントを設定する必要があります。その後はすべての見積りをベースラインと比較して行うべきです。

1 つのユーザー ストーリーに何ポイントのストーリー ポイントを割り当てるべきかについて、厳格な規則はありません。まったく同じストーリーであっても、ポイント数はチームによって異なる場合があります。重要なことは、設定したベースラインに従って各チームが一貫してストーリー ポイントを割り当てるようにすることです。

## 電子署名 / 変更理由の入力とトラッキング

280 ページの「電子署名と変更理由の設定」の説明にあるように、電子署名 あるいは 変更理由の設定を行った場合、ある特定のステータスから別の特定のステータスにアイテムが変わったときに、電子署名または変更理由を入力しなければなりません。

たとえば、アイテムのステータスが「Draft」から「Approved」が変わったときに電子署名が必要かもしれません。

### 電子署名の追加

電子署名を入力するには、次の操作を行います。

1. 証明書のチェックボックスをオンにします。
2. ユーザー名とパスワードを入力します。
3. [Submit] ボタンをクリックします。

**Parasoft E-Signature** ✕

**電子署名の理由:**  
要求 ステータス "Review" から "Finalized" に変更

私は以前に、この電子署名が、合衆国の食品医薬品局の連邦規則コード  
タイトル 21 パート 11 のとおり、私の手書きの署名と法的に結び付けられた  
封等物であるという証明書に署名し提出しました。

私は上記の声明が真実であることを証明し、この電子署名を私の手書きの  
署名の代わりに使用することに同意します。

ユーザー:

パスワード:

入力した電子署名は、[ 変更履歴 ] タブの下部にある [ 電子署名 ] の表に記録されます。

#### 電子署名

1 アイテム				
ユーザー	日付	フィールド名	古い値	新しい値
admin	2012-10-17 13:52:30	ステータス	Review	Finalized

## カスタム電子署名

カスタム電子署名を追加するには、[ 電子署名の設定 ] タブで [ カスタム ] オプションを選択し、テキストを入力します。

The screenshot shows the PARASOFT Concerto Project Center interface. The top navigation bar includes 'Main', 'Projects', and 'Settings' tabs. Below this, the 'E-Signature / Notifications Settings' section is active, with sub-tabs for 'E-Signature Events', 'E-Signature Settings', and 'Event Notifications'. The 'E-Signature Dialog Message' section has two radio buttons: 'FDA (Food and Drug Administration)' and 'Custom'. The 'Custom' option is selected. Below the radio buttons is a text input field containing the text 'My own custom message'. A 'Save' button is located at the bottom of this section.

The screenshot shows a dialog box titled 'Parasoftware E-Signature'. It contains the following elements:
 

- Reason for E-Signature:** Scenario Run Status change from "Ready for Review" to "Approved."
- My own custom message:** A section with a checkbox and the text: "I certify that the above statement is true and agree to use this electronic signature in place of my written signature."
- User:** A text input field.
- Password:** A text input field.
- Buttons:** 'Submit' and 'Cancel' buttons at the bottom.

## 変更理由

変更理由を入力するには、次の操作を行います。

1. なぜ現在の変更を行ったか、その理由を入力します。

**Modification Reason** ✕

要求 ステータスを "Review" から "Finalized" に変更する理由を指定してください。

変更の理由:

Submit
Cancel

2. [Submit] ボタンをクリックします。

入力した変更理由は、[ 変更履歴 ] タブの一番上にある [ 変更履歴 ] の表に記録されます。

変更履歴

変更されたフィールド	削除された値	追加された値
変更時刻: [2012-10-17 13:57:55] by admin		
ステータス	Review	Finalized
変更の理由		実装が完了したため
説明	The search for the products should be: - intuitive - fast - results should be informative. To achieve it we should have: - quick search always visible - detailed search - mouse over details info for each item on the search result list	The search for the products should be: - intuitive - fast - results should be informative. To achieve it we should have: - quick search always visible - detailed search - mouse over details info for each item on the search result list

# Project Center の要求と欠陥 / 改善

このセクションの内容：

- 要求の作成と管理
- 欠陥 / 改善
- 要求と欠陥 / 改善のタスク化

## 要求の作成と管理

Project Center では、一括して要求を定義、インポート、管理、レビューできるだけでなく、コード、テスト、タスク、および欠陥 / 改善と相互に関連付けることもできます。要求は、ユーザーが手動で追加することも、.xls ファイルなどの他のソースからインポートすることもできます。

要求を管理する機能には [ 要求 ] ドロップダウン メニューからアクセスできます。

このセクションの内容：

- 要求の定義
- 要求のインポート
- 要求の削除
- 要求の検索
- 要求の削除
- 要求の一括更新
- 要求と他の成果物との関連付け
- 要求レポートの参照

## 要求の定義

1. [ 要求 ] ドロップダウン メニューの [ 新規 ] をクリックします。[ 新規要求 ] ページが表示されます。

 新規要求

 全般

プロジェクト: *	利用可能なプロジェクト		選択されたプロジェクト
	Default Project Empty Project MyProj1	すべて追加 >> 追加 削除 すべて削除 <<	FDAdemo
名前: *	<input type="text"/>		
説明:	<input type="text"/>		
ランク: *	<input type="text" value="50000"/>		
オリジナル ID:	<input type="text"/>		
ステータス: *	Review <input type="button" value="▼"/>		

2. [ 利用可能なプロジェクト ] フィールドのプロジェクト名をクリックして選択し、[ 追加 ] をクリックします。[ すべて追加 ] をクリックして、要求をすべてのプロジェクトの成果物とすることもできます。
3. 要求の名前を入力します。
4. 説明 ( オプション ) に要求の詳細な説明を入力し、[ 作成 ] をクリックします。

## 要求定義のフィールド

フィールド	説明
プロジェクト	この要求が関連付けられる Development Testing Platform プロジェクト。0 個、1 個、または複数のプロジェクトを指定できます。
名前	( 必須 ) この要求を簡潔に表す名前。
説明	この要求についての詳しい説明。
重要度	この要求の優先度に割り当てる数字。 最高：1-19999 高：20000-39999 中：40000-59999 低：60000-79999 最低：80000-99999
オリジナル ID	要求管理システムで使用される ID。 同じ ID を持つ要求を複数のプロジェクトで使用できます。たとえばオリジナル ID = M の要求 X がプロジェクト Y にある場合、オリジナル ID = M の要求 Z をプロジェクト Z で追加できます。ただし、1 つのプロジェクトが同じ ID の要求を複数持つことはできません。
期限	この要求を実装する期限。
担当者	この要求を担当するユーザーのログイン名。
ストーリー ポイント	特定の機能を実装するために必要な作業量を見積るために使用される単位。詳細については 73 ページの「ストーリー ポイント」を参照してください。
親要求 ID	この要求の親要求の ID。詳細については 82 ページの「親子の関係の指定」を参照してください。

**既存の要求の子である要求を新規に定義する**

既存の要求の子である要求を新規に定義することもできます。82 ページの「子の指定」を参照してください。

---

## 要求への添付ファイルの追加

既存の要求に添付ファイルを追加するには、次の操作を行います。

1. 添付ファイルを追加する要求の [ 要求の編集 ] ページを開きます。
2. [ 全般 ] タブで [ 添付 ] セクションまでスクロールします。
3. [ 追加 ] をクリックし、添付ファイルを追加します。

---

## 要求からの添付ファイルの削除

1. 添付ファイルを削除する要求の [ 要求の編集 ] ページを開きます。
2. [ 全般 ] タブで [ 添付 ] セクションまでスクロールします。
3. 削除する添付ファイルを選択して [ 削除 ] をクリックします。

**MySQL データベースのデフォルトの最大パケット サイズは 1MB です**

このデフォルト設定のために、添付ファイルの最大ファイル サイズは 1MB に制限されます。MySQL データベースの最大パケット サイズを変更せずに 1 MB より大きいファイルを添付しようとすると、エラーが発生する可能性があります。

---

## 要求へのコメントの追加

既存の要求にコメントを追加するには、次の操作を行います。

1. 要求の [ 要求の編集 ] ページを開きます。
2. [ コメント ] タブをクリックします。
3. [ 新規コメント ] フィールドにコメントを入力して [ コメントの追加 ] をクリックします。



**成果物へのリンクの追加**

Development Testing Platform の成果物（要求、タスク、欠陥、シナリオ、シナリオの実行、テストセッション）へのリンクを追加するには、次の書式を使用します。

- @task #
- @req #
- @pr #
- @scenario #
- @scenario run #
- @test session #

@ 記号は任意です。エンティティ名と ID の間に、空白以外の文字を入れてはいけません。

たとえば、次のコメントはタスク 123 にリンクします。

- This requirement should extend the functionality implemented in @task 123.

**親子の関係の指定**

要求を作成または編集するときに、要求の親子関係を指定できます。

**親の指定**

現行の要求に対して親を指定するには、次の操作を行います。

1. [ 新規要求 ] または [ 要求の編集 ] ページで [ 関係 ] セクションまでスクロールします。
2. フィールドに入力するか [ 検索 ] リンクをクリックして、親要求の ID を指定します。

**子の指定**

子にする要求がすでに作成済みである場合、その要求を既存の要求の子として指定できます。そうでない場合、子にする要求を作成するときに親子関係を定義できます。

**既存の要求の場合**

「既存の要求」が「現行の要求の子」であることを指定するには、次の操作を行います。

1. 親要求の [ 要求の編集 ] ページで [ 関係 ] セクションまでスクロールします。
2. [ 割り当て ] ボタンをクリックします。

- 表示されたダイアログで子要求を選択します ( 左端のカラムのアイコンをクリックします )。

検索の詳細: プロジェクト: FDAdemo; ステータス: Finalized;			
5 アイテム			
選択	ID	名前	説明
<input checked="" type="checkbox"/>	1	ボタンオン後2秒以内に初期状態を表示する	
<input type="checkbox"/>	3	アイドル状態から10秒後にシャットダウンする	
<input type="checkbox"/>	4	体温を0.1度Cの精度で、測定する	
<input type="checkbox"/>	5	体温を測定15秒後に計算し、アラームを鳴らし、表示する	

### 新規要求の場合

「既存の要求の子」として新規に要求を作成するには、次の操作を行います。

- 次のいずれかの操作を行います。
  - 親要求の [ 要求の編集 ] ページで、[ 関係 ] セクションまでスクロールして [ 新規 ] をクリックします。
  - [ 要求の管理 ] ページで、親要求を右クリックし、ショートカットメニューから [ 要求に子を追加 ] を選択します。
- 通常と同じように残りのフィールドを定義します。

### 親子関係の削除

親子関係を削除するには、次の操作を行います。

- 親の [ 要求の編集 ] ページを開きます。
- 削除する子関係を選択します。
- [ 割り当て解除 ] をクリックします。

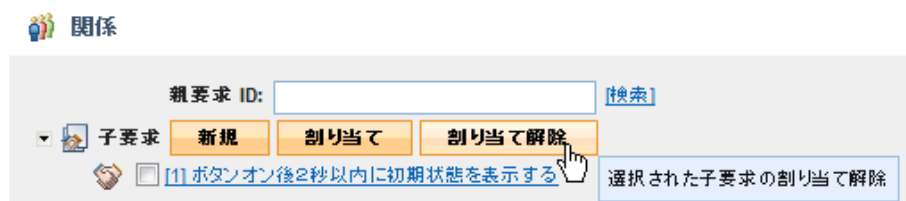


図 44: レビュー ステータスのモニタリング

## 要求のインポート

要求は、csv、xls、および xlsx ファイルからインポートすることができます。csv ファイル中で要求の名前などに National 文字を使用している場合、Development Testing Platform にインポートする前に、必ず UTF-8 エンコーディングでファイルを保存してください。

ファイルから要求をインポートするには、次の操作を行います。

1. [要求] ドロップダウン メニューの [インポート] をクリックします (または要求ページの上の [インポート] アイコンをクリックします)。
2. [ステップ 1: ターゲット プロジェクトの選択] セクションで、要求が属するプロジェクトを指定します。0 個、1 個、または複数のプロジェクトを指定できます。
3. [ステップ 2: ファイルの選択 (.csv, .xls, .xlsx)] セクションで [ファイルの選択] ボタンをクリックします。ファイル選択ダイアログで適切なファイルを選択して [開く] ボタンをクリックします。
4. [ステップ 3: マッピングの構成] セクションで、ファイル中の異なる列を Development Testing Platform の要求フィールドにマッピングします。必須なのは [名前] フィールドだけです。なお、インポートされる要求は、次のいずれかの方法で階層 (親子関係) を表現できます。
  - オリジナル ID (オプション) で、PARENT\_ID.CHILD\_ID という形式で階層関係を表します。たとえば、ある要求のオリジナル ID が 10.1 の場合、この要求には親要求があり、親要求のオリジナル ID が 10 であることを意味します。
  - 親 ID とオリジナル ID を使って親子関係を指定します。親 ID を指定する場合、オリジナル ID も指定しなければなりません。

📁 ファイルから要求をインポート

ファイルパス:<sup>\*</sup>

+ ファイルの選択

要件一覧-Concerto5\_0.xlsx

終了

ステップ 3: マッピングの構成

インポートされたフィー<sup>\*</sup>

Concerto 要求 フィールド	選択されたファイルからのカラム名
名前	名前
オリジナル ID	---
説明	1
ランク	2
ステータス	5
期限	7
オーナー	
ストーリー ポイント	

次へ >>      リセット

### カスタム フィールドのインポート

277 ページの「カスタム フィールドの使用」で説明しているように、管理者によって追加されたカスタム フィールドをインポートしたりマッピングすることができます。

5. [次へ] をクリックします。
6. 提案されたマッピングをレビューします。
7. 提案されたマッピングで問題ない場合、[Submit] をクリックします。問題がある場合、[Back] をクリックして修正します。

オプションを設定して、インポートされる要求と既存の要求が重複する場合に、新しい要求で既存の要求を置き換えるか、それともインポートを止めるかを指定します。

📁 ファイルから要求をインポート

既存の要求との競合:

ID: 36  
 名前: UIの改善  
 オリジナル ID: 1  
 説明: UIを刷新する

アクション  
 インポートしない  
 上書き

変更されたフィールド	削除された値	追加された値
ランク	30000	10000

## 要求の検索

要求を検索するには、次の操作を行います。

1. [ 要求 ] ドロップダウン メニューの [ 検索 ] をクリックするか、任意の要求ページの [ 検索 ] アイコンをクリックします。  
 [ 要求の検索 ] ページが表示されます。

全般

ID: <input type="text"/>	オリジナル ID: <input type="text"/>
名前: <input type="text"/>	説明: <input type="text"/>
実装: <input type="text" value="任意"/>	コメント: <input type="text"/>
ランク範囲下限: <input type="text"/>	タスクとしてスケジュール: <input type="text" value="任意"/>
期間終了: <input type="text"/>	期限範囲開始: <input type="text"/>
オーナー: <input type="text"/>	期間終了: <input type="text"/>
プロジェクト: <input type="text" value="Default Project&lt;br/&gt;Empty Project&lt;br/&gt;FDAdemo&lt;br/&gt;MyProj1"/>	ステータス: <input type="text" value="Review&lt;br/&gt;Finalized"/>
作成者: <input type="text"/>	ストーリー ポイント: <input type="text"/>
親 ID: <input type="text"/>	

要求の作成/変更

作成日: <input type="text"/>	修正日: <input type="text"/>
<input type="text"/>	<input type="text"/>

2. 以下の 1 つ以上の条件を入力します。80 ページの「要求定義のフィールド」にあるフィールドに加えて、次の条件で検索を実行できます。
  - **ID:** 要求に割り当てられた ID 番号。
  - **オリジナル ID:** オリジナルの RMS での要求の ID。

- **実装**：要求の実装のステータス。
- **タスクとしてスケジュール**：この要求に対して Project Center タスクが作成されたかどうか。
- **ステータス**：要求の現在のステータス。
- **オーナー**：要求を作成したユーザーのログイン名。
- **作成日**：要求が Development Testing Platform に追加された日付。
- **修正日**：Development Testing Platform で要求が変更された日付。

3. [ 検索 ] ボタンをクリックします。

検索結果が表形式で表示されます

🔍 要求

検索の詳細: 制約なし

32 アイテム タスクとしてスケジュール 印刷 一括更新 再スキャン

<input type="checkbox"/>	ID	オリジナル ID	名前	説明	ランク	ステータス	親	#欠陥
<input type="checkbox"/>	<a href="#">1</a>		ボタンオン後2秒以内に初期状態を表示する		50000 - M	Finalized	<a href="#">2</a>	0
<input type="checkbox"/>	<a href="#">2</a>		LCD表示中にボタンをオンするとシャットダウンする		50000 - M	Finalized		0
<input type="checkbox"/>	<a href="#">3</a>		アイドル状態から10秒後にシャットダウンする		50000 - M	Finalized		1
<input type="checkbox"/>	<a href="#">4</a>		体温を0.1度Cの精度で、測定する		50000 - M	Finalized		0
<input type="checkbox"/>	<a href="#">5</a>		体温を測定15秒後に計算し、アラームを鳴らし、表示する		30000 - H	Finalized		0
<input type="checkbox"/>	<a href="#">9</a>		要求1		50000 - M	Review		0
<input type="checkbox"/>	<a href="#">10</a>		要求1 - 子1		50000 - M	Review	<a href="#">9</a>	0
<input type="checkbox"/>	<a href="#">11</a>		孫要求1		50000 - M	Review	<a href="#">10</a>	0

この表には次の情報が表示されます。

- **ID**：要求に割り当てられた ID 番号。ID をクリックすると [ 要求の編集 ] ページが開き、選択した要求の詳細情報にアクセスできます。
- **オリジナル ID**：オリジナルの RMS での要求の ID。
- **説明**：要求についての簡潔な説明。
- **ランク**：要求の重要度。
- **親**：要求の親の ID 番号。このカラムが空白の場合、その要求には親要求がありません。
- **# 欠陥**：要求に関連付けられた欠陥、および要求の実装中に発見された欠陥の数。
- **実装の進捗**：実装の進捗。要求に関連付けられた、クローズのタスクの割合が表示されます。この値は次の式で計算されます。

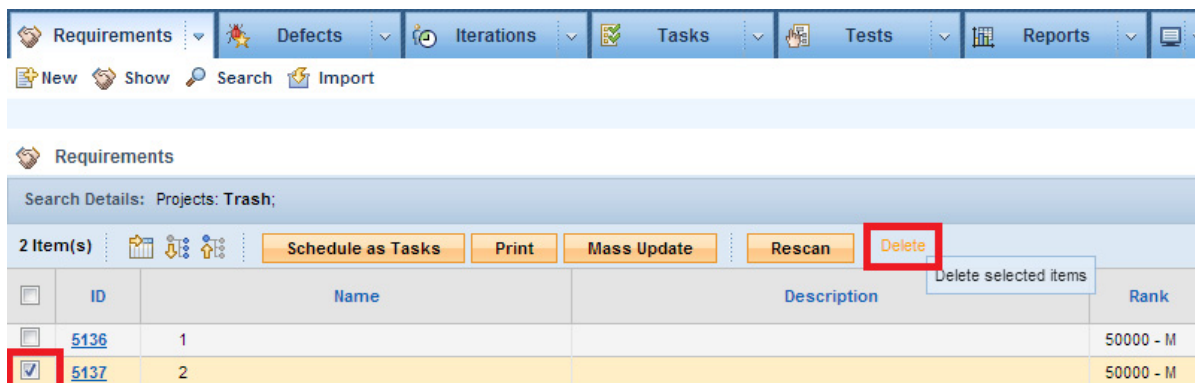
完了のタスクの数 / タスクの総数 ( キャンセルされたタスクは含まない ) \* 100.

- **タスク**：要求に関連付けられたタスクの数がステータス ( オープン、作業中、完了、キャンセル ) ごとに表示されます。

## 要求の削除

要求を恒久的に削除すると、その要求とすべての子要求が Development Testing Platform から削除され、元に戻すことはできません。

1. Project Center メニューの [ 要求 ] > [ 表示 ] をクリックします。
2. 要求を選択し、メニューの [ 削除 ] リンクをクリックします。



3. Development Testing Platform からの恒久的な削除に関する確認が表示されたら、要求の削除を確定させます。

### 注意

- 要求の削除の確認は 2 回行う必要があります。
- 制限モードでは、[ 削除 ] リンクは表示されません。

## 要求の編集

Project Center では、いつでも各要求の詳細情報を参照したり編集することができます。要求に関連付けられたタスクと欠陥 / 改善だけでなく、要求がいつ誰によって作成されたかについての情報も含まれます。要求の詳細情報は [ 要求の編集 ] ページに表示されます。表示されている要求 ID をクリックすることで、[ 要求の検索 ] などのさまざまなページから [ 要求の編集 ] ページを開くことができます。

[ 要求の編集 ] ページの機能は以下のページ タブに分割されています。

- 全般
- タスク
- 欠陥
- 変更履歴
- コード
- 自動テスト
- シナリオ

要求を作成した後、要求のフィールドを更新すると、自動的に電子メールによる通知が行われます。詳細については 272 ページの「タスクと欠陥の通知メールの設定」を参照してください。

## 全般

[全般] タブには、選択された要求についての全般的な情報が表示されます。情報に添付ファイルがある場合、添付ファイルも表示されます。

## コメント

[コメント] タブでは、要求についてのコメントを入力およびレビューするだけでなく、要求レビューの開始とモニタリングを行うことができます。


要求のレビューの詳細については 83 ページの「レビュー ステータスのモニタリング」を参照してください。

コメントの入力の詳細については 81 ページの「要求へのコメントの追加」を参照してください。

## タスク

[タスク] タブには、選択した要求に関連付けられたタスクのリストが表示されます。タスクは、要求に関連する欠陥 / 改善から直接要求に関連付けることができます。言い換えれば、ある欠陥にタスクが関連付けられていて、その欠陥がある要求に関連付けられている場合、そのタスクはこの要求の [タスク] タブに表示されます。

タスクと要求の関係は、タスク リストの [関係] 列で見ることができます。

 要求の編集: [25] 本体の組み上げ

全般									
全般	コラボレーション	タスク  4	欠陥  0	変更履歴	コード  0	自動テスト  0	シナリオ  2/2		
4 アイテム									
<span>新規タスク</span> <span>割り当て</span> <span>割り当て解除</span> <span>一括更新 </span>									
<input type="checkbox"/>	ID	名前	ステータス	テスト	優先度	反復	オーナー	関係	
<input type="checkbox"/>	<a href="#">35</a>	要件 25 の実装: 本体の組み上げ	完了	成功	高	<a href="#">8</a>	nagata	要求 <a href="#">25</a>	
<input type="checkbox"/>	<a href="#">152</a>	Requirement Review: 本体の組み上げ	完了	不要	高		nagata	要求レビュー <a href="#">25</a>	
<input type="checkbox"/>	<a href="#">153</a>	Requirement Feedback Review: 本体の組み上げ	キャンセル	不要	高		admin	要求レビュー <a href="#">25</a>	
<input type="checkbox"/>	<a href="#">154</a>	Requirement Review: 本体の組み上げ	オープン	不要	高		nagata	要求レビュー <a href="#">25</a>	

[タスク] タブでは、選択した要求に対するタスクを作成、割り当て、または変更することができます。

選択した要求に対して新規にタスクを作成するには、[新規タスク] ボタンをクリックします。

**注意:** 特定の要求に対して新規にタスクを作成する他の方法については「要求と欠陥 / 改善のタスクとしてのスケジューリング」を参照してください。

選択した要求に既存のタスクを割り当てるには、[Assign Existing Task(s)] ボタンをクリックします。

## 欠陥

[欠陥] タブには、選択した要求に関連する欠陥の詳細が表示されます。



要求が実装されているが、実装中でバグが発見された場合、バグ トラッキング システムに問題レポートが送られて Development Testing Platform にインポートされるべきです。そうすれば、Project Center 中の要求とこの欠陥を手動で関連付けることができます。

要求を欠陥 / 改善や他の成果物と関連付ける方法については、71 ページの「欠陥、改善、テスト、タスクの関連付け」を参照してください。

**注意：**欠陥に関連付けられたタスクが [欠陥] タブに表示されている場合、「タスク」で説明しているように、これらのタスクは選択された要求の [タスク] タブにも表示されます。

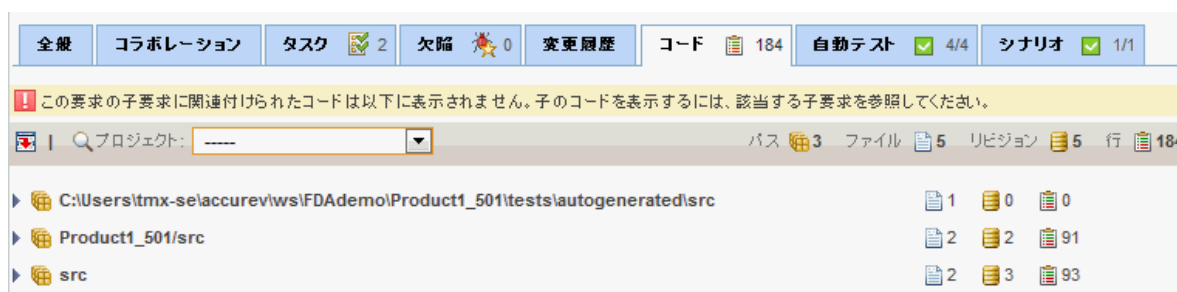
## 変更履歴

[変更履歴] タブには、この要求に対するすべての変更が表示されます。

## コード

[コード] タブには、選択した要求の実装に関連付けられたすべてのファイル リビジョンが表示されます。

図 45: [コード] タブ



選択した要求に関連付けられたすべてのファイル リビジョンが階層ツリー形式で表現されます。リビジョンごとに、作成者、日付、および変更された行数が表示されます。

要求とコードを関連付ける方法については 70 ページの「ソース コードと成果物の関連付け」を参照してください。

## 自動テスト

[自動テスト] タブには、選択した要求をテストするために実行されたすべての自動テストのレポートが表示されます。

このレポートの詳細については 204 ページの「自動テストの詳細」を参照してください。

自動テストと要求を関連付ける方法については 60 ページの「要求とテスト ケースとの関連付け」を参照してください。

## シナリオ

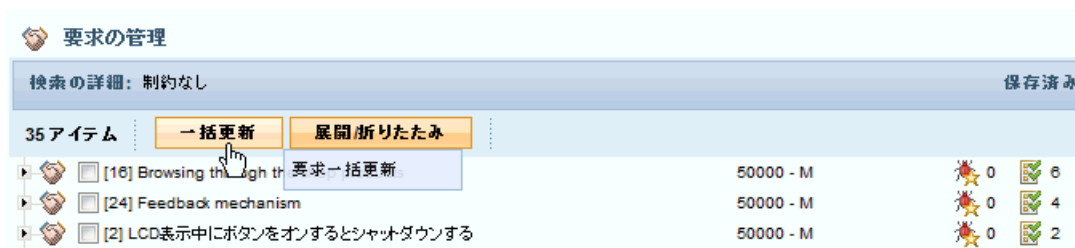
[シナリオ] タブには、選択した要求に対するすべてのシナリオとシナリオの実行についての情報が表示されます。シナリオの名前、関係、ステータスなどの詳細情報が表示されます。

シナリオがどのように要求や他の成果物と関連するかについては、71 ページの「欠陥、改善、テスト、タスクの関連付け」を参照してください。

## 要求の一括更新

[ 要求の管理 ] ページなどの要求リストから一度に複数の要求を更新するには、次の操作を行います。

1. 更新する要求のチェックボックスをそれぞれオンにします。
2. [ 一括更新 ] ボタンをクリックします。



3. 変更を行って [ 保存 ] をクリックします。

### 更新する選択済み要求

3 アイテム

ID	名前	ランク	ステータス	組	オーナー	期限	プロジェクト
16	Browsing through the shop products	50000 - M	Review				MyProj1
2	LCD表示中にボタンをオンするとシャットダウンする	50000 - M	Finalized		nagata		FDAdemo
24	Feedback mechanism	50000 - M	Review				MyProj1

### 一括更新のスコープ

選択された要求の子孫に変更を伝播

### 全般

プロジェクトの変更

プロジェクト:

ランクの変更

ランク:

オーナーの変更

オーナー:

期限の変更

期限:

---

## 要求と他の成果物との関連付け

70 ページの「Project Center 成果物（欠陥、要求、タスク、テスト）の関連付け」で説明しているように、タスク、欠陥、およびテストを要求と相互に関連付けることができます。

---

## 要求レポートの参照

176 ページの「要求レポート」を参照してください。

---

## 欠陥 / 改善

Project Center では、一括して欠陥 / 改善を定義、インポート、管理、レビューするだけでなく、コード、テスト、タスク、欠陥 / 改善と相互に関連付けることができます。統合されたバグ トラッキング システムまたは Development Testing Platform の内部バグ トラッキング システムから欠陥を追加できます。

このセクションの内容：

- Development Testing Platform による欠陥管理の容易化
- Development Testing Platform の内部バグ トラッキング システムの使用
- 欠陥の検索 (基本検索)
- 欠陥の検索 (詳細検索)
- 欠陥 / 改善の編集
- 欠陥 / 改善の一括更新
- 欠陥 / 改善と他の成果物との関連付け
- 欠陥 / 改善レポートの参照
- 欠陥 / 改善のオンデマンドでの再スキャン

---

## Development Testing Platform による欠陥管理の容易化

Development Testing Platform でプロジェクトの欠陥管理を容易にする方法は 2 種類あります。

- 既存のバグ トラッキング システム (JIRA、Buzgilla、ClearQuest、HP QC) をすでに使用している場合、そのバグ トラッキング システムを Development Testing Platform に統合できます。詳細については 336 ページの「バグ トラッキング システムおよび要求管理システムの統合」を参照してください。
- Development Testing Platform の内部バグ トラッキング システムを使用できます。詳細については 96 ページの「Development Testing Platform の内部バグ トラッキング システムの使用」を参照してください。

どちらの方法でも次の操作を行うことができます。

- 欠陥 / 改善に関連するソース コードをトラッキングする。
- 欠陥 / 改善に関連する自動テストと手動テストをモニタリングする
- 欠陥と改善を要求に関連付ける
- 欠陥と改善をタスクとしてスケジューリングする

---

## 要求のトレーサビリティ

サポートされるバグ トラッキング システムを Development Testing Platform に統合して使用する場合でも、Development Testing Platform の内部バグ トラッキング システムを使用する場合でも、欠陥に対して高度なトレーサビリティを得ることができます。Development Testing Platform

は欠陥からテスト、ソース コード、および要求への関連付けをトラッキングします。Development Testing Platform は履歴トレンド レポートも出力します。これらのアイテムとその傾向を分析すると、効率的に「継続か中止か」を判断し、事後分析を行うのに役立ちます。

- 欠陥レポートがプロジェクト中の欠陥の傾向をレポートします。詳細については以下を参照してください。
  - 144 ページの「欠陥および改善レポート」
  - 187 ページの「[ 欠陥 / 改善のステータス ] レポート」
  - 186 ページの「[ 欠陥 / 改善の統計 ] レポート」
- 修正された欠陥に関連付けられたソース コードのトレーサビリティ。詳細については以下を参照してください。
  - 36 ページの「欠陥の修正」
  - 101 ページの「欠陥 / 改善の編集」の「コード」
- 修正された欠陥の品質を保証するために実行されるテストのトレーサビリティ。詳細については以下を参照してください。
  - 36 ページの「欠陥の修正」
  - 37 ページの「結果のテスト」
  - 101 ページの「欠陥 / 改善の編集」の「自動テスト」および「シナリオ」
- 欠陥から要求へのトレーサビリティ。
- 欠陥に関連するタスクの作業。詳細については以下を参照してください。
  - 114 ページの「イテレーションの計画」
  - 106 ページの「要求と欠陥 / 改善のタスクとしてのスケジューリング」
  - 115 ページの「タスクとしてのバックログ項目のスケジューリング」

---

## タスクと欠陥の同期化

Development Testing Platform の内部バグ トラッキング システムを使用する場合でも、サポートされるバージョンの Bugzilla、HP Quality Center、または JIRA と Development Testing Platform を統合して使用する場合でも、関連するバグ トラッキング システム中の課題にステータスの変更とコメントを伝播できます。

Development Testing Platform 中の関連タスクが作成または変更されると、Development Testing Platform は自動的に次の情報をバグ トラッキング システム中の欠陥に追加します。

- **タスクの作成:** バグ トラッキング システム中の特定の欠陥に関連付けてタスクを作成する場合、BTS Updater はこの欠陥に " この課題は Parasoft Development Testing Platform に Task としてスケジュールされています。 " というコメントを追加します。

例 :

- ▼  tmx がコメント - 17/10/12 3:40 午後 を追加  
この課題はタスク 103 として Parasoft Concerto にスケジュールされました。  
[http://DELL-I3100-11.tmx.local:80/grs/jsf/planning/task/edit\\_task.jsf?entityId=103](http://DELL-I3100-11.tmx.local:80/grs/jsf/planning/task/edit_task.jsf?entityId=103)

これは、Bugzilla、HP Quality Center、JIRA、および Development Testing Platform の内部バグ トラッキング システムについてサポートされます。

- **タスク ステータスの変更** : Project Center でこの タスクのステータスが (たとえば "オープン" から "進行中" に) 変わる場合、バグ トラッキング システム中の関連する課題のステータスも更新されます。次の表は、Project Center のステータスと対応するバグ トラッキング システムのステータスをまとめたものです。

Project Center	Bugzilla	HP Quality Center
オープン	reopened	new
進行中	assigned	open
キャンセル	fixed/won't fix	rejected
完了	fixed/resolved	fixed

これは、Bugzilla、HP Quality Center、および Development Testing Platform の内部バグ トラッキング システムについてサポートされます。

- **コメントの追加** : Development Testing Platform のタスクにコメントが追加される場合、Project Center は関連するバグ トラッキング システムの課題にもコメントを追加します。

これは、Bugzilla、HP Quality Center、JIRA、および Development Testing Platform の内部バグ トラッキング システムについてサポートされます。

この同期を有効にするには、[バグ トラッキング システムの更新] オプションを有効にして更新したタスクを保存する必要があります。

保存
 バグ トラッキング システムの更新

Development Testing Platform の内部バグ トラッキング システムとの同期化については 97 ページの「欠陥とタスクの同期化の構成」を参照してください。

既存の外部トラッキング システムとの同期化については 226 ページの「バグ トラッキング システムとの同期」を参照してください。

## Development Testing Platform の内部バグ トラッキング システムの使用

Development Testing Platform をバグ トラッキング システムとして使用できます。

Development Testing Platform は次の機能を提供します。

- 一般的な バグ トラッキング システムの機能。欠陥と改善を編集、管理、検索する機能など。
- 高度な欠陥トレーサビリティ (93 ページの「Development Testing Platform による欠陥管理の容易化」で説明)。

欠陥に関連するすべての機能は [欠陥] タブにあります。

Development Testing Platform は、統合されたバグ トラッキング システム中の関連する課題にステータスの変更とコメントを伝播します。Development Testing Platform 中の関連タスクが作成または変更されると、Development Testing Platform は自動的に次の情報を欠陥に追加します。

- **タスクの作成** : バグ トラッキング システム中の特定の欠陥に関連付けてタスクを作成する場合、BTS Updater はこの欠陥に「この課題は Parasoft Development Testing Platform に Task としてスケジュールされています。」というコメントを追加します。
- **タスク ステータスの変更** : Project Center でこの タスクのステータスが (たとえば "オープン" から "進行中" に) 変わる場合、バグ トラッキング システム中の関連する課題のステータスも更新されます。
- **コメントの追加** : Development Testing Platform のタスクにコメントが追加される場合、Project Center は関連するバグ トラッキング システムの課題にもコメントを追加します。

### 欠陥の追加

Project Center で欠陥 / 改善を定義する基本操作は以下のとおりです。

1. [欠陥] ドロップダウン メニューから [新規] を選択します。[新規欠陥 / 改善] ページが表示されます。
2. [タイプ] ドロップダウン リストから [欠陥] または [改善] を選択します。
3. [全般] セクションで詳細を定義します。各フィールドについては以下の「欠陥の定義のフィールド」を参照してください。

### 欠陥の定義のフィールド

フィールド	説明
プロジェクト	欠陥が関連付けられる Development Testing Platform プロジェクト。0 個、1 個、または 2 個以上のプロジェクトを指定できます。
名前	欠陥を簡潔に説明する名前。このフィールドは必須です。

フィールド	説明
説明	欠陥についての詳しい説明。
ステータス	欠陥の現在のステータス。 欠陥が、有効な解決されたステータス (たとえばデフォルトでは Resolved または Verified) のいずれかを持つ場合： <ul style="list-style-type: none"> <li>欠陥は "Not tested/Test failed/Fixed" カテゴリに入ります (どれに入るかはこの欠陥に関連付けられたテストによって決まります)。</li> <li>欠陥 / 改善の統計 レポートおよび 欠陥の検索 での欠陥の古さは、「欠陥が作成されたとき」から「欠陥が解決されたとき」までの経過時間として計算されます。このレポートの詳細については 187 ページの「[ 欠陥 / 改善のステータス ] レポート」を参照してください。</li> </ul>
優先度	この欠陥を修正する優先度。たとえば、できるだけ早く修正しなければならない欠陥には高い優先度を設定します。欠陥を記録したただですぐに修正する予定がないのであれば、低い優先度を設定します。
重要度	この欠陥の重要度。たとえば、主要な機能の使用を妨げている欠陥には高い重要度を設定します。機能に影響しない欠陥 (GUI 上での色の不一致など) には低い重要度を設定します。
予定時間	要求を実装する最終期限。
ストーリー ポイント	特定の機能を実装するために必要な作業量を見積るために使用される単位。詳細については 73 ページの「ストーリー ポイント」を参照してください。
オーナー	要求を担当するユーザーのログイン名。
監視者	要求の実装のモニタリングを担当するユーザーのログイン名。
コンポーネント	この欠陥が存在する製品コンポーネント。
アクション	この欠陥に関連して取られたアクション。
検出バージョン	この欠陥が検出された製品のバージョン。
クローズ バージョン	この欠陥が解決された製品のバージョン。

## 欠陥とタスクの同期化の構成

94 ページの「タスクと欠陥の同期化」で説明しているように、Development Testing Platform は関連するバグ トラッキング システムの課題にステータスの変更とコメントを伝播することができます。

タスクのステータスが変更されると、TaskBTSUpdate.xml ファイルで定義されているように欠陥のステータスが変更されます。このファイルは grs/config の下にあります。デフォルトではファイルの内容は次のとおりです。



```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <open bts-status="New"/>
  <in-progress bts-status="Assigned"/>
  <completed bts-status="Resolved"/>
  <canceled bts-status=""/>
  <esig-workflow ignore="false"/>
</configuration>
```

タスクのステータスが open に変更されると、欠陥のステータスが new に変更されます。タスクのステータスが in-progress に変更されると、欠陥のステータスが assigned に変更されます。以下 completed と canceled も同様に変更されます。

最期の行 <esig-workflow ignore="false"/> は、欠陥のステータスが電子署名または変更理由を必要とする場合、欠陥のステータスが変更されないことを表します。電子証明または変更理由を必要とせずに欠陥のステータスを変更したい場合はこの行を次のように変更します。

```
<esig-workflow ignore="true"/>
```

---

## 欠陥への添付ファイルの追加

既存の欠陥に添付ファイルを追加するには、次の操作を行います。

1. 添付ファイルを追加する欠陥の [ 欠陥の編集 ] ページを開きます。
2. [ 全般 ] タブで [ 添付 ] セクションまでスクロールします。
3. [ 追加 ] をクリックし、添付ファイルを追加します。

後で添付ファイルを削除するには、次の操作を行います。

1. 添付ファイルを削除する欠陥の [ 欠陥の編集 ] ページを開きます。
2. [ 全般 ] タブで [ 添付 ] セクションまでスクロールします。
3. 削除する添付ファイルを選択して [ 削除 ] をクリックします。

### MySQL データベースのデフォルトの最大パケット サイズは 1MB です

このデフォルト設定のために、添付ファイルの最大ファイル サイズは 1MB に制限されます。MySQL データベースの最大パケット サイズを変更せずに 1 MB より大きいファイルを添付しようとする、エラーが発生する可能性があります。

---

## 欠陥の検索 ( 基本検索 )

欠陥に対して基本的な検索を実行するには、次の操作を行います。


1. [ 欠陥 ] > [ 検索 ] をクリックします。[ 欠陥 / 改善 検索 ] ページが表示されます。
2. 全般的な検索のために、以下の 1 つ以上の検索条件を入力します。標準のフィールドに加えて、次のフィールドに注意してください。

- **タスク名** 検索される欠陥 / 改善が関連付けられているタスク。言い換えると、欠陥 / 改善がスケジュールリングされたタスク。
- **要求の名前** 検索される欠陥 / 改善が関連付けられている要求。言い換えると、実装中に欠陥が発見された要求。
- **タスクとしてスケジュール** 要求のために Project Center のタスクがすでに作成済みかどうか。
- **プロジェクト** 検索される欠陥 / 改善が属している Development Testing Platform プロジェクト。
- **解決済み** 欠陥 / 改善が " 解決済み " としてマークされているかどうか。
- **欠陥トラッキング システム** 100 ページの「欠陥の検索 ( 詳細検索 )」を参照。

3. [ 検索 ] をクリックします。

検索結果が [ 欠陥 / 改善 ] ページに表として表示されます ( 図 46)。

図 46: 欠陥 / 改善の検索結果


 欠陥改善

検索の詳細: プロジェクト: PolicyCdemo;

54 アイテム タスクとしてスケジュール 一括更新 再スキャン

<input type="checkbox"/>	ID	名前	ステータス	優先順位
<input type="checkbox"/>	1 	webページの「戻る」ボタンで失われる部分があ	ASSIGNED	P2

この表には以下の情報が表示されます。

- **チェックボックス**: 検索結果で複数の行を選択するために使用します。選択された欠陥 / 改善はタスクとしてスケジュールリングできます。それには [ タスクとしてスケジュール ] をクリックします。スケジュールのダイアログが表示され、選択された欠陥 / 改善ごとにタスクが自動的に作成されます。
- **ID**: 次のリンクがあります。
  - **ID**: [ 欠陥 / 改善の編集 ] ページが開き、選択した項目の詳細が表示されます。
  - **BTS Link**  : バグ トラッキング システムのページが開き、対応する項目が表示されます ( バグ トラッキング システムに欠陥 / 改善を参照するための Web インターフェイスがある場合 )。
- **システム**: 欠陥 / 改善を Project Center と統合したバグ トラッキング システム。
- **タイプ, 説明, 優先順位, 重要度, ステータス**: バグ トラッキング システムからインポートされた標準的な欠陥 / 改善のプロパティ。
- **関連タスク**: 特定の欠陥 / 改善がスケジュールリングされたタスク。
- **関連要求**: 特定の欠陥 / 改善が関連付けられている要求。言い換えると、実装で欠陥が発見された要求、または実装で改善が提示された要求。

## 欠陥の検索 ( 詳細検索 )

詳細検索セクションでは、より細かく欠陥 / 改善の検索条件を指定できます。指定できる検索条件は、システムのベンダーによって異なります。

欠陥 / 改善の詳細検索を実行するには、次の操作を行います。

1. Project Center で [ 欠陥 ] ドロップダウン メニューの [ 検索 ] をクリックします。  
[ 欠陥 / 改善 検索 ] ページが表示されます。
2. [ 欠陥トラッキング システム ] ドロップダウン リストから欠陥トラッキング システムを選択します ( 図 47)。

図 47: [ 欠陥トラッキング システム ] ドロップダウン リスト

### 欠陥トラッキングシステム固有の属性

選択したバグ トラッキング システムが、残りの検索プロパティを決定します。この検索プロパティはベンダーに固有のもので、たとえば Bugzilla を選択した場合 ( 図 48)、Bugzilla に固有の条件選択フィールドが表示されます ( 製品、ステータス、解決、優先度、重要度、ハードウェア、および OS)。

図 48: 検索プロパティ

3. [ 検索 ] ボタンをクリックします。  
図 46 に似た表に検索結果が表示されます。

## 欠陥 / 改善の編集

Project Center では、いつでも個々の欠陥 / 改善の詳細情報を参照したり編集することができます。欠陥 / 改善に関連付けられたタスクと要求の情報も含まれます。これらの詳細情報は [欠陥 / 改善の編集] ページに表示されます。表示されている要求 ID をクリックすることで、[欠陥 / 改善 検索] などのさまざまなページから [欠陥 / 改善の編集] ページを開くことができます。

[欠陥 / 改善の編集] ページの機能は以下のページ タブに分割されています。

- 全般
- タスク
- 要求
- 変更履歴
- コード
- 自動テスト
- シナリオ

## 全般

[全般] タブには、選択された欠陥 / 改善についての全般的な情報が表示されます。説明、コメント、タイプなどです。情報に添付ファイルがある場合、添付ファイルも表示されます。ストーリー ポイントを参照 / 変更することもできます。詳細については 73 ページの「ストーリー ポイント」を参照してください。

図 49: [全般] タブ

欠陥/改善の編集: [10] 表示の一部が欠けている

全般 タスク 0 要求 0 変更履歴 コード 0 自動テスト 0 シナリオ 0 シナリオ実行 0

ステータス: Assigned

説明およびコメント (2) 添付 (0)

説明およびコメント

表示部右上のアイコンの一部が欠けています。

コメント #1 コメント者 admin 日時 2012-10-16 17:31

新アイコンリソースファイルに変更予定です。

新規コメント:

添付ファイル

0 アイテム 追加 保存

プロジェクト: FDAdemo

名前: 表示の一部が欠けている

タイプ: 欠陥

優先度: Medium

重要度: Medium

コンポーネント: ---

アクション: ---

検出バージョン: ---

クローズ バージ...: ---

オーナー: nagata

監視者:

作成者: admin

### 成果物へのリンクの追加

Development Testing Platform の成果物（要求、タスク、欠陥、シナリオ、シナリオの実行、テストセッション）へのリンクを追加するには、次の書式を使用します。

- @task #
- @req #
- @pr #
- @scenario #
- @scenario run #
- @test session #

@ 記号は任意です。エンティティ名と ID の間に、空白以外の文字を入れてはいけません。

たとえば、次のコメントはタスク 123 にリンクします。









- This requirement should extend the functionality implemented in @task 123.

## タスク

[ タスク ] タブには、選択した欠陥 / 改善に関連付けられたタスクのリストが表示されます。

図 50: [ タスク ] タブ

🔥 欠陥/改善の編集: [3] UI改善要望

全般	タスク  1	要求  1	変更履歴	コード  1	自動テスト  0	シナリオ  0	シナリオ実行  1
1 アイテム	新規タスク	割り当て	割り当て解除	一括更新 			
<input type="checkbox"/>	ID	オリジナル ID	名前 		ステ		
<input type="checkbox"/>	<a href="#">98</a>		FR 3 の実装: 改善要望		作業中		

[ タスク ] タブでは、選択した欠陥 / 改善に対するタスクを作成、検索、管理、またはインポートすることができます。

選択した欠陥 / 改善に対して新規にタスクを作成するには、[ タスク ] ドロップダウン メニューの [ 新規 ] をクリックします。

**注意:** 特定の欠陥 / 改善に対して新規にタスクを作成する他の方法については「要求と欠陥 / 改善のタスクとしてのスケジューリング」を参照してください。

## 要求

[ 要求 ] タブには、選択した欠陥 / 改善に関連する要求の詳細が表示されます。

🔥 欠陥/改善の編集: [3] UI改善要望

全般 タスク 1 要求 1 変更履歴 コード 1 自動テスト 0 シナリオ 0

1 アイテム

<input type="checkbox"/>	ID	名前
<input type="checkbox"/>	3	アイドル状態から10秒後にシャットダウンする

この欠陥 / 改善を要求と関連付けるには、[ 割り当て ] ボタンをクリックします。

この欠陥 / 改善を要求との関連付けを解除するには、[ 割り当て解除 ] ボタンをクリックします。

欠陥 / 改善と要求を関連付ける他の方法については 71 ページの「欠陥、改善、テスト、タスクの関連付け」を参照してください。

## 変更履歴

[ 変更履歴 ] タブには、この欠陥に対するすべての変更が表示されます。

## コード

[ コード ] タブには、選択した欠陥 / 改善のソース コードに関連付けられたすべてのファイル リビジョンが表示されます。

図 51: [コード] タブ

🔥 欠陥/改善の編集: [3] UI改善要望

全般 タスク 1 要求 1 変更履歴 コード 1 自動テスト 0 シナリオ 0 シナリオ実行 1

🔍 プロジェクト: FDAdemo

📁 Sensor1002/src

- 📄 BaseDisplay.cxx 1 1
  - 📄 Demo\_dev2/3 diff FR3 +1/-0 naga

選択した欠陥 / 改善に関連付けられたすべてのファイル リビジョンが階層ツリー形式で表現されます。リビジョンごとに、作成者、日付、および変更された行数が表示されます。

---

## 自動テスト

[ 自動テスト ] タブには、選択した欠陥 / 改善のために生成されたすべての自動テストのレポートが表示されます。

このレポートの詳細については 204 ページの「自動テストの詳細」を参照してください。

自動テストと欠陥 / 改善を関連付ける方法については「欠陥 / 改善とテスト ケースとの関連付け」を参照してください。

---

## シナリオ

[ シナリオ ] タブには、選択した欠陥 / 改善に対して作成されたすべてのシナリオとシナリオの実行についての情報が表示されます。シナリオの名前、関係、ステータスなどの詳細情報が表示されません。

シナリオがどのように欠陥 / 改善に関連するかについては、71 ページの「欠陥、改善、テスト、タスクの関連付け」を参照してください。

---

## 欠陥 / 改善の一括更新

[ 欠陥 / 改善 ] ページなどの欠陥リストで一度に複数の欠陥を更新するには、次の操作を行います。

1. 更新する欠陥のチェックボックスをそれぞれオンにします。
  2. [ 一括更新 ] をクリックします。
  3. 変更を行って [ 保存 ] をクリックします。
- 

## 欠陥 / 改善と他の成果物との関連付け

70 ページの「Project Center 成果物 ( 欠陥、要求、タスク、テスト ) の関連付け」で説明しているように、タスク、要求、およびテストを欠陥 / 改善と相互に関連付けることができます。

---

## 欠陥 / 改善レポートの参照

186 ページの「欠陥 / 改善レポート」を参照してください。

---

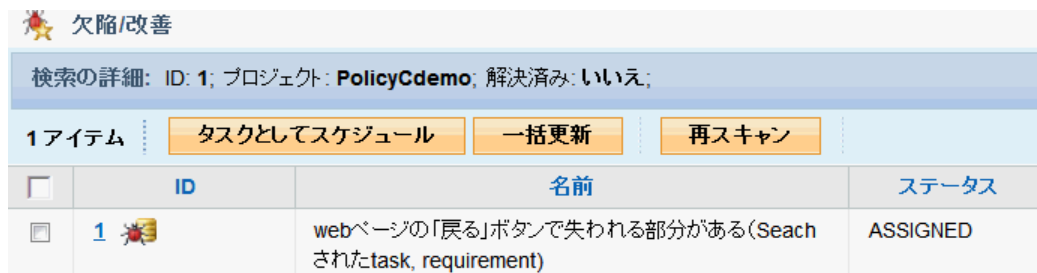
## 欠陥 / 改善のオンデマンドでの再スキャン

Development Testing Platform を外部のバグ トラッキング システムと統合した場合、Development Testing Platform は 15 分ごとにバグ トラッキング システムに自動的に接続し、バグ トラッキング システムからのデータを更新します。バグ トラッキング システムの項目を更新したばかりで、その変更をすぐに Development Testing Platform に反映したい場合、オンデマンドで同期化することができます。

欠陥と改善をオンデマンドで再スキャンするには、次の操作を行います。

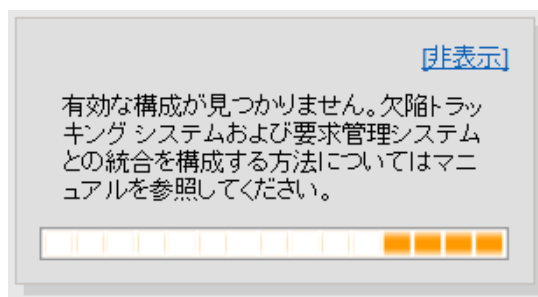
1. Project Center のホーム ページで [ 欠陥 ] ドロップダウン メニューの [ 検索 ] をクリックします。
2. [ 欠陥 / 改善 ] ページで [ 再スキャン ] ボタンをクリックし、再スキャンする欠陥 / 改善をクリックします ( 図 52)。

図 52: 欠陥 / 改善 - 欠陥 / 改善の再スキャン



Project Center がバグ トラッキング システムに接続し、欠陥 / 改善のデータを更新します。スキャンが進行中であることを表すダイアログが表示されます ( 図 53)。

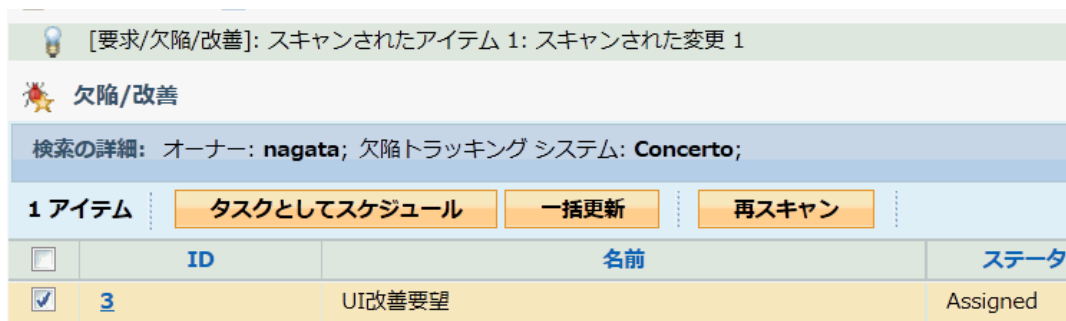
図 53: スキャン進行中のダイアログ



3. ( オプション ) 再スキャンのプロセスをバックグラウンドで実行しつつ Development Testing Platform の使用を続けるには、[ 非表示 ] をクリックします。

再スキャンが完了すると、[ 欠陥 / 改善 ] ページの上部にメッセージが表示されます ( 図 54)。

図 54: スキャン完了メッセージ





## 要求と欠陥 / 改善のタスク化

要求が Project Center に格納されたら、マネージャーは作業の計画と分配を行うことができます。効率的に作業を分配して開発の進捗をモニタリングするために、要求は複数のタスクに分割することができます。

同様に、マネージャーは開発者、QA チーム、顧客からレポートされた特定の欠陥 / 改善に対してタスクを作成できます。

次の方法で要求と欠陥 / 改善のためのタスクを作成できます。

- 要求と欠陥 / 改善のタスクとしてのスケジューリング
- 要求と欠陥 / 改善のタスクの手動作成
- [ タスク ] タブからのタスクの新規作成

**注意：** [ 予定 ] ページでも要求と欠陥 / 改善のタスクを作成できます。それには、バックログ アイテムをターゲットのイテレーションにドラッグします。詳細については 115 ページの「タスクとしてのバックログ項目のスケジューリング」を参照してください。

## 要求と欠陥 / 改善のタスクとしてのスケジューリング

要求と欠陥 / 改善は、自動生成された条件を使ってタスクとしてスケジューリングすることができます。

要求または欠陥 / 改善をタスクとしてスケジューリングするには (1 つの要求につき 1 つのタスク、または 1 つの欠陥 / 改善につき 1 つのタスクを生成するには)、次の操作を行います。

**注意：** 以下の説明は要求をタスクとしてスケジューリングするための手順ですが、欠陥 / 改善の場合も手順は同様です。

1. 実装する要求を検索します。
  - a. [ 要求 ] ドロップダウン メニューの [ 検索 ] をクリックします。
  - b. [ 要求の検索 ] ページで、既知の条件 ( プロジェクトなど ) を入力し、[ 検索 ] をクリックして結果を表示します。
2. タスクとしてスケジュールする要求のチェックボックスをオンにし、[ タスクとしてスケジュール ] ボタンをクリックします。

### タスクとしてスケジュール

プロジェクトまたはイテレーション ID を要求タスクにまだ割り当てていない場合、そのことをリマインドするダイアログが表示されます。[OK] をクリックします。

[ 要求のスケジュール ] ページが表示され、選択した要求ごとにタスク名の提案が表示されます。

図 55: [ 要求のスケジュール ] ページ

全般

プロジェクト: demo

反復 ID:

タスクリスト

2 アイテム				
名前	Related to	オーナー	予定時間	
FR 8 の実装: 画像がきれいにしたい	FR8	<input type="text"/> <input type="button" value="表示"/> <input type="button" value="除去"/>	1d	
FR 7 の実装: 不要なグラフを隠す機能が欲しい	FR7	<input type="text"/> <input type="button" value="表示"/> <input type="button" value="除去"/>	1d	

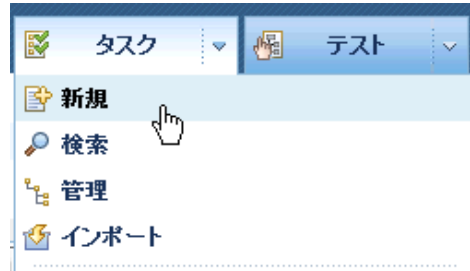
このページには、タスクが属するイテレーションのプロジェクトと ID が表示されるほか、次のフィールドが表示されます。

- 名前: 実行されるタスクの名前
  - オーナー: タスクの完了に責任があるチームメンバーの名前。
  - 予定作業時間: タスクの完了に必要な作業日数の予測。
3. ターゲット プロジェクトとイテレーション ID を選択 / 追加します。
  4. 次の項目を指定します。
    - オーナー: タスクの完了に責任があるチームメンバー (複数指定可能)。
    - 予定作業時間: タスクの完了に必要な作業日数の予測。
  5. [ 作成 ] をクリックします。

## 要求と欠陥 / 改善のタスクの手動作成

要求または欠陥 / 改善ごとに、複数の詳細なタスクを手動で作成するには、次の操作を行います。

1. [ タスク ] ドロップダウン メニューの [ 新規 ] を選択します。



[ 新規タスク ] ページが表示されます。

### 新規タスク

関係: ---

全般

プロジェクト: PolicyCdemo 反復 ID: [検索] [追加] [ビュー]

優先度: 中 監督者: [表示] [削除]

オーナー: [表示] [削除]

名前: \*

説明:

許可されるリンクの書式: http|ftp|https://company.com, www.company.com, @task NNN, @req NNN (@task 111, @REQ 222)

タグ: このエンティティに対して定義されたタグはありません [タグの追加]

テストが必要:

タイプ: 開発

2. 次のいずれかの操作を行います。

- 要求のタスクを作成するには、[ 関係 ] ドロップダウン リストから [ 要求の実装 ] を選択して [ 要求 ID ] を指定します。

関係: 要求の実装 要求 ID: \* [検索] [ビュー]

- 欠陥 / 改善のタスクを作成するには、[ 関係 ] ドロップダウン リストから [ 欠陥 / 改善の修正 ] を選択して [ 欠陥 / 改善 ID ] を指定します。

関係:  欠陥/改善 ID:

- [ 全般 ] セクションの [ 名前 ] フィールドにタスクの名前を入力します。
- [ 作成 ] をクリックします。

**注意：** 選択した要求に対してタスクはいくつでも作成できます。

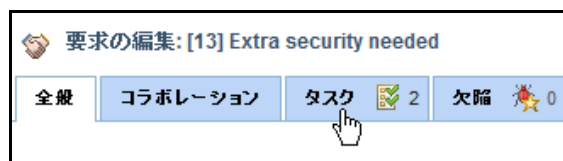
特定の要求または欠陥 / 改善に割り当てられているタスクの数とそのステータス ( オープン、作業中、完了、失敗、キャンセル など ) を調べるには、対応する [ 要求の検索結果 ] の表を参照します。

## [ タスク ] タブからのタスクの新規作成

タスクの作成と割り当ては、[ タスク ] タブから行うこともできます。[ タスク ] タブは [ 要求の編集 ] および [ 欠陥 / 改善の編集 ] ページに表示されます。

[ タスク ] タブからタスクを新規に作成するには、次の操作を行います。

- 次のいずれかの操作を行います。
  - [ 要求の編集 ] ページを表示します。
  - [ 欠陥 / 改善の編集 ] ページを表示します。
- [ タスク ] タブをクリックします。



- [ 新規タスク ] ボタンをクリックします。



[ 新規タスク ] ウィンドウが表示されます。

関係: 要求の実装  要求 ID: 13 [検索](#) [ビュー](#)

**全般**

プロジェクト: MyProj1  反復 ID:  [検索](#) [追加](#) [ビュー](#)

優先度: 中  監督者:  [表示](#) [追加](#)

オーナー:  [表示](#) [追加](#)

名前: 要求 [IDENTIFIER] の実装 : [SUMMARY]

説明:

許可されるリンクの書式: http|ftp|https://company.com, www.company.com, @task NNN, @req NNN (@task 111, @REQ 222)

タグ: このエンティティに対して定義されたタグはありません  [タグの追加](#)

テストが必要:

タイプ: Development

**作業時間**

予定

開始日:   予定時間:

[作成](#)

タスク タイプ、エンティティ ID、および名前が提案されて表示されます。また、[テストが必要] チェックボックスがデフォルトでオンに設定されます。

4. (オプション) 必要に応じて提案された値を変更します。
5. (オプション) 次の項目を指定します。
  - オーナー: タスクの完了に責任があるチーム メンバー (複数指定可能)。
  - 予定作業時間: タスクの完了に必要な作業日数の予測。
6. [作成] をクリックします。

# Project Center のタスクとイテレーション

このセクションの内容：

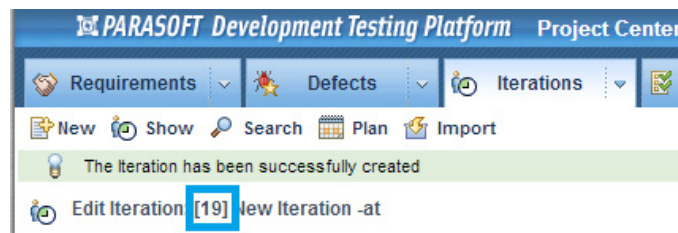
- イテレーションへのプロジェクトの分割
- イテレーションの計画
- タスクの新規作成
- タスクの管理
- CSV からのタスクのインポート
- 予測作業時間の変更
- タスクの検索
- イテレーションの編集
- イテレーション内のタスクの管理

# イテレーションへのプロジェクトの分割

イテレーションとは、チームが作業の計画と組み立てを行う際に役立つ開発サイクルを表します。スクラムなど一部のアジャイル開発手法では、イテレーションは「スプリント」と呼ばれます。一般的なスプリントの期間は 2 週間ですが、イテレーションの長さは任意です。アジャイル開発プロセスを実践していない組織でも、プロジェクトを複数のイテレーションに分割するのは、作業を管理するうえで有効な方法です。

1. [イテレーション] > [新規] をクリックします。
2. ドロップダウンメニューからプロジェクトを選択します。デフォルトで現在のプロジェクトが選択されています。
3. イテレーションの名前を入力します。
4. 説明、スケジュールされたリソース、および開始日および終了日は任意オプションです。
5. [作成] をクリックして新しいイテレーションを保存します。

イテレーションが作成されると、ID が生成されて割り当てられます。



## プロジェクトをイテレーションに分割するその他の方法

- [予定] ページで、ターゲットのイテレーションにバックログ項目をドラッグ アンド ドロップして、プロジェクトをイテレーションに分割できます。詳細については 114 ページの「イテレーションの計画」を参照してください。

- Project Center で作成するほかに、MS Project からイテレーションをインポートすることもできます。詳細については 207 ページの「Microsoft Project の統合」を参照してください。



# イテレーションの計画

[ 予定 ] ページでは、プロジェクト バックログ ( オープンの要求、欠陥、タスクなど ) を参照し、どのイテレーションで何をすべきなのかを計画し、必要に応じて計画を見直して修正することができます。

[ 予定 ] ページは 2 つのセクションに分かれています。左側の [ プロジェクト バックログ ] と右側の [ イテレーション ] です ( 図 56 )。

図 56: プロジェクト バックログとイテレーション

The screenshot shows the Jira '予定' (Forecast) page. On the left, the 'プロジェクト バックログ' (Project Backlog) section is active, displaying a list of 29 items. The items are sorted by rank and include details like ID, name, rank, and points. On the right, the 'イテレーション' (Iterations) section shows a list of 10 items, including milestones and tasks with their start and end dates.

ID	名前	期限	ランク	ストーリー	ポイント
[95]	Browsing through the shop products		50000		
[103]	Feedback mechanism		50000		
[100]	Shopping cart and purchase procedure		50000		
[112]	Support for credit cards		50000		
[108]	Tools for administration		50000		
[91]	User registration and profile		50000		
[49]	レゴロボット作成		50000		

[ プロジェクト バックログ ] セクションには、特定のプロジェクトまたは指定されたカスタム フィルターに対するバックログ項目 ( 最大で 500 個までの欠陥 / タスク ) が表示されます。バックログ項目には以下が含まれることがあります。

- 要求 ( 統合された要求管理システムから必要に応じてインポート )
- 欠陥 / 改善 ( 統合されたシステムからインポート )
- タスク ( まだイテレーションに割り当てられていない Development Testing Platform タスク )

[ イテレーション ] セクションには、特定のプロジェクトに存在するイテレーションが最大で 10 個まで表示されます。

イテレーションの計画には次のアクティビティが含まれます。

- プロジェクト バックログ項目の参照
- ストーリー ポイントの割り当て ( オプション )
- タスクとしてのバックログ項目のスケジューリング
- チーム メンバーによるタスクのグループ化
- スケジューリングしたタスクの管理

## プロジェクト バックログ項目の参照

バックログ項目を参照するには、参照するバックログ項目の種類に対応するタブをクリックします。たとえば、プロジェクト バックログ中の欠陥を参照するには、[ 欠陥 ] タブをクリックします。

**注意：** Project Center での参照 / 検索の詳細については「Project Center での検索操作」を参照してください。

デフォルトでは、プロジェクト バックログにはデフォルト プロジェクトからのオープンなバックログ項目が表示されます。

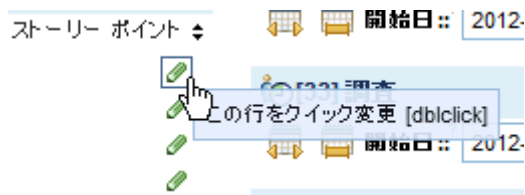
- **要求：** デフォルト プロジェクトに属している要求で、実装されてなく、まだタスクとしてスケジューリングされていない要求。
- **欠陥：** バグ トラッキング システムの項目で、デフォルト プロジェクトに属していてバグ トラッキング システムで解決されてなく、まだタスクとしてスケジューリングされていない項目。
- **タスク：** デフォルト プロジェクトに属しているオープンなタスクで、他のイテレーションに属していないタスク。

## ストーリー ポイントの割り当て (オプション)



アジャイルのストーリー ポイントをプロジェクト バックログ中の各項目に割り当てることができます。イテレーションにバックログ項目を割り当てる場合、その項目のストーリー ポイントはそれぞれ 1 つの予定された作業日に変換されます。

バックログ項目にストーリー ポイントを割り当てるには、次の操作を行います。

1. [クイック変更] ボタンをクリックします。



2. ストーリー ポイントの数を指定します。たとえば 5.0 など。
3. 次のいずれかの操作を行います。

- 確定するには、[変更をコミット] ボタン  をクリックします。
- キャンセルするには [変更をキャンセル] ボタン  をクリックします。

## タスクとしてのバックログ項目のスケジューリング

[予定] ページでは、バックログ項目をタスクとしてスケジューリングすることができます。タスクとしてバックログ項目をスケジューリングすると、それは特定のイテレーションに割り当てられません。

タスクとしてバックログ項目をスケジューリングするには、次の操作を行います。

- 1つのバックログ項目をプロジェクト バックログから 1つのイテレーションにドラッグ アンド ドロップします。



- [ドラッグ アンド ドロップ] ボタン をクリックし、左側の [バックログ] セクションから右側の [イテレーション] セクションにバックログ項目をドラッグします。

**注意：** 選択したバックログ項目でストーリー ポイントが定義されている場合、その対応する日数が、作成されるタスクに割り当てられます。たとえば、選択したバックログ項目に 5.0 ストーリー ポイントが定義されている場合、タスク候補の [予定時間] は 5d に設定されます。

タスク候補が作成されます。

**注意：** タスク候補は、ID を持たないタスクです。なぜなら、まだタスクとして確定されていないからです。

	ID	名前	ステータス	優先度	S.P.	
	<a href="#">57</a>	プロジェクト登録のページの遷移数を減らす	オープン	中		
	<a href="#">58</a>	検索スピードを倍にする	オープン	最高		
	<a href="#">59</a>	SOAtestとの機能テスト連携動作を確認する	オープン	最高		

- (オプション) タスク候補を編集してその [オーナー] または [優先度] を変更します。
- [保存] をクリックします。

タスク候補がタスクとして確定されます。[イテレーション] セクションで行った変更が保存されます。

## チームメンバーによるタスクのグループ化

プロジェクト チームのメンバーは、イテレーション中のタスクをオーナーでグループ化することができます。

- [イテレーション] ドロップダウン メニューの [予定] をクリックして [予定] ページに移動します。

[予定] ページの左側はプロジェクト バックログを表し、右側は現行のイテレーションを表します (図 56)。

- 右側の [イテレーション] セクションで、[詳細の表示] アイコンをクリックしてイテレーションを参照します。



そのイテレーションのタスクが表示されます。


図 57: 選択したイテレーションのタスク リスト

[30] Milestone 2: Basic functionality implementation 66d

開始日: 2012-08-26 終了: 2012-09-25 Dev.#: 3.0

12アイテム

	ID	名前	オーナー	予定	残り	優先度	
<input checked="" type="checkbox"/>	119	Implement 要件 113 : Support for AmEx	admin	7d		中	↕
<input checked="" type="checkbox"/>	120	Implement 要件 114 : Support for Visa	JohnSmith	6d		中	↕
<input checked="" type="checkbox"/>	121	Implement 要件 112 : Support for credit cards	DavidScott	5d		中	↕
<input checked="" type="checkbox"/>	122	Implement 要件 96 : Related Products	admin	5d		中	↕
<input checked="" type="checkbox"/>	123	Implement detailed search	JohnSmith	5d		中	↕
<input checked="" type="checkbox"/>	124	Automated notification	JohnSmith	5d		中	↕
<input checked="" type="checkbox"/>	125	Implement 要件 98 : Site Map	admin	2d		中	↕
<input checked="" type="checkbox"/>	126	Implement 要件 99 : Support for categories and subcategories	DavidScott	5d		中	↕

3. 表の右上の [ チーム メンバーごとのグループ化 ]  をクリックしてオーナーでタスクをグループ化します。タスク オーナーのリストが表示されます ( 図 58)。オーナーの名前の

左側にある矢印をクリックすると、選択したイテレーションにおけるそのオーナーのタスクリストを参照できます。

図 58: タスク オーナーのリスト

<<オーナーのいないタスク>>

▼ DavidScott :: 見残り残作業時間合計: 16d

4アイテム

	ID	名前	予定	残り	優先度
<input checked="" type="checkbox"/>	121	Implement 要件 112 : Support for credit cards	5d		中
<input checked="" type="checkbox"/>	126	Implement 要件 99 : Support for categories and subcategories	5d		中
<input checked="" type="checkbox"/>	127	Implement 要件 107 : Functionality of giving rating for the product	3d		中
<input checked="" type="checkbox"/>	130	Implement 要件 104 : Newsletter Signup	3d		中

▶ JohnSmith :: 見残り残作業時間合計: 19d

▶ admin :: 見残り残作業時間合計: 19d

## スケジューリングしたタスクの管理

新規にイテレーションを作成してバックログ項目をタスクとしてスケジューリングした後、スケジューリングしたタスクをイテレーション内とイテレーション間の両方で管理したい場合があります。スケジューリングしたタスクの管理には、次のアクティビティが含まれます。

- イテレーション キャパシティの分析
- イテレーションのタスクの参照
- 作業負荷の調整

## イテレーション キャパシティの分析

イテレーション キャパシティ バーは、「スケジューリングされた作業量」と「まだ使用できる自由時間の量」をチーム リーダーに通知します (図 59)。

図 59: イテレーション キャパシティ バー

[31] Milestone 3: Advanced functionality implementation and release

開始日: 2012-09-26 終了: 2012-10-26 Dev.#: 3.0

経過時間: 42d/69d

イテレーション キャパシティ バーの長さは、イテレーション キャパシティの合計を就業日の日数で表します。イテレーション キャパシティは「イテレーション中の就業日の日数」と「そのイテレーションに割り当てられたチーム メンバーの人数」を乗算して決定されます。

**例：**図 59 は、2010-11-03 から 2010-11-24 の期間を示します。イテレーションの指定された期間において 16 日の就業日（週末を除いた暦日）が与えられます。このイテレーションを担当するチームメンバーは 4 名です。したがって、「16 × 4 人日 = 64」のイテレーション キャパシティが与えられます。

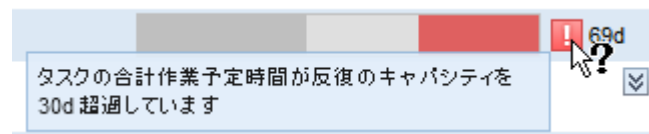
イテレーション キャパシティ バーには、イテレーションのスケジューリングされたリソースについての情報が表示されます。

- **経過時間：**すでに経過した「人日」の数。まだ開始していないイテレーションの場合、この値はゼロです。
- **スケジュール済み：**このイテレーションに割り当てられた、すべてのオープンなタスクの予定時間の合計。

**注意：**タスクを作成するときに「予定時間」を定義した場合、イテレーション中でそのタスクがスケジューリングされる時間は自動的に更新されます。これはイテレーション キャパシティ バーに反映されます。

- **空き時間：**スケジューリングしてタスクを追加できる、まだ空いている「人日」の数。
- **超過：**イテレーション キャパシティが許す数よりも多くのタスクがスケジューリングされている場合、このステータスが表示されます（図 60）。

図 60: 超過したイテレーション

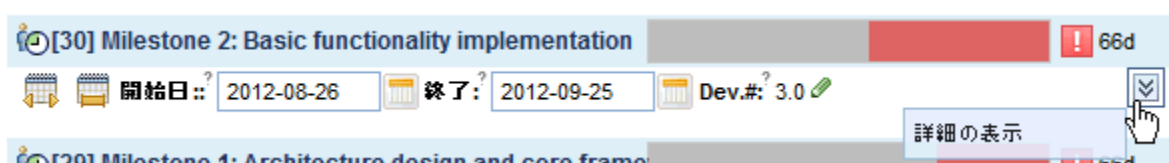


## イテレーションのタスクの参照

[イテレーション] セクションでは、イテレーション内のすべてのタスクについての基本情報を参照することもできます。

あるイテレーションに対してスケジューリングしたタスク（およびタスクの詳細）を参照するには、次の操作を行います。

- 個々のイテレーションの [詳細の表示] アイコンをクリックします。



タスク リストが展開し、そのイテレーションに対してスケジューリングされたすべてのタスクが表示されます ( 図 61)。

図 61: 展開されたイテレーションタスク リスト

	ID	名前	オーナー	予定	残り	優先度
	<a href="#">131</a>	Implement 要件 92 : Extra security needed - HTTPS Support	admin	3d		中
	<a href="#">132</a>	Extra security needed - Automated logout after 5 minutes idle time	JohnSmith	4d	4d	中
	<a href="#">133</a>	Implement 要件 111 : Flexible look and feel	DavidScott	10d		中
	<a href="#">134</a>	Customizable items layout on the page	JohnSmith	6d		中

タスク リストには、各タスクの基本情報が表示されます。ID、名前、オーナー、予定 [ 時間 ]、残り [ 時間 ]、優先度などです。

**注意：**

- タスクのプロパティの詳細については「タスクの新規作成」を参照してください。
- タスク プロパティの変更の詳細については「タスクの管理」を参照してください。

あるイテレーションに対してスケジューリングしたタスクを非表示にするには、次の操作を行います。

- [ 詳細の非表示 ] アイコンをクリックしてタスク リストを閉じます。  
タスクが閉じます。

## 作業負荷の調整

イテレーション キャパシティ バーを分析し、各イテレーションでスケジューリングされたタスクを参照することによって、イテレーションごとの作業量をよりの確に把握できます。次の操作を行って作業負荷のバランスを取る必要があるかどうかを判断できます。

- タスクの移動
- イテレーションからのタスクの削除

どちらの操作もドラッグ アンド ドロップで行うことができます。どの程度の作業が新たにスケジューリングされたか、あるいはスケジュールから外されたかがチーム メンバー全員に分かるように、これらの操作の結果がイテレーション キャパシティ バーに反映されます。

## タスクの移動

[イテレーション] セクションでは、イテレーション間でタスクを移動することができます。たとえば図 60 のように、あるイテレーションでタスクが過密な場合、自由時間がある別のイテレーションにタスクを移動できます。

イテレーション間でタスクを移動するには、次の操作を行います。

1. イテレーションの [詳細の表示] アイコンをクリックします。

イテレーションのタスクリストが表示されます。

**注意:** スケジューリングされたタスクの参照については 119 ページの「イテレーションのタスクの参照」を参照してください。

2. 重なった矢印のアイコンを使って、タスクをドラッグ アンド ドロップして別のイテレーションに移動します (図 62)。

図 62: イテレーション間でのタスクの移動

ID	名前	オーナー	予定	残り	優先度
131	Implement 要件 92 : Extra security needed - HTTPS Support	admin	3d		中
132	Extra security needed - Automated logout after 5 minutes idle time	JohnSmur	4d	4d	中

タスクが別のイテレーションに移動されます。

## イテレーションからのタスクの削除

[イテレーション] セクションでは、スケジューリングしたタスクをイテレーションから削除することもできます。図 60 のようにイテレーションが過密になった場合やタスクが不要になった場合、タスクを削除すると便利です。

イテレーションからタスクを削除するには、次の操作を行います。

1. イテレーションの [詳細の表示] アイコンをクリックします。

イテレーションのタスク リストが表示されます。

**注意:** スケジューリングされたタスクの参照については 119 ページの「イテレーションのタスクの参照」を参照してください。



2. 削除するタスクのチェックボックスをオフにします (図 63)。

図 63: タスクの削除の確認

The screenshot shows a task management interface for a milestone titled "[31] Milestone 3: Advanced functionality implemen". It displays a table of tasks with columns for ID, Name, Owner, Estimated, Remaining, and Priority. Task 131, "Implement 要件 92: Extra security needed - HTTPS", is selected with a mouse cursor over its unchecked checkbox. Task 132, "Extra security needed - Automated logout after 5", is checked. A tooltip "包含 / 除外" is shown over the checkbox for task 131.

ID	名前	オーナー	予定	残り	優先度
131	Implement 要件 92: Extra security needed - HTTPS	admin	3d		中
132	Extra security needed - Automated logout after 5	JohnSmith	4d	4d	中

変更の保存についての通知が [ イテレーション ] セクションの上に表示されます。

3. [ イテレーション ] セクション上部の [ 保存 ] をクリックします (図 64)。

図 64: タスクの削除の確定

The screenshot shows the top of the task management interface. A notification bar is visible with a "保存" (Save) button. A tooltip "すべての変更をデータベースに格納" (Save all changes to the database) is shown over the button. The interface also shows a search bar and a "4 アイテム" (4 items) indicator.

チェックボックスがオフのタスクがイテレーションから削除されます。

---

## タスクの新規作成

1. [タスク] > [新規] をクリックして [新規タスク] ページを開きます。
2. タスクの詳細な情報を入力し (123 ページの「タスクの詳細」を参照)、[作成] をクリックします。

タスクを作成するその他の方法については、124 ページの「新規タスクを作成するその他の方法」を参照してください。

---

## タスクの詳細

- 関係：特定の要求または欠陥 / 改善にタスクが関連するかどうかを指定します (関係を指定する場合、要求または欠陥 / 改善の ID が必要です)。たとえば、「開発」、「設計 / 計画」「品質」、「レビュー」といったさまざまなタイプの複数のタスクが特定の 1 つの要求に関連することもあります。
- プロジェクト (必須)：タスクが属するプロジェクトを選択します。
- イテレーション ID：このタスクが属するイテレーションの ID を入力します。
- 優先度：ドロップダウン リストからタスクの優先度を選択します。[最高]、[高]、[中]、[低]、[最低] の中から選択します。
- 監督者：タスクのステータスについて電子メールで連絡を受け取るべきであるすべてのユーザー。
- オーナー：タスクに割り当てられ、タスクを完了することを期待されるチーム メンバー。
- 名前：タスクを簡潔に説明する名前。
- 説明：このタスクを完了するために必要なことについての説明。
- タグ：このタスクに関連付けるタグを入力します。
- テストが必要：タスクのテストが必須であるか、任意であるかを指定します。
- 親 ID：新規タスクの親とする別のタスクの ID を入力します。
- 先行：新規タスクの前に完了している必要があるタスクの ID (複数可) を入力します。
- タイプ：次のいずれかを選択します。
  - スタンドアロン：要求リストにない機能要求の実装にタスクが関連する可能性が高いことを示します。また、他のオプションに当てはまらないタスク タイプを示すこともできます。
  - 設計 / 計画：アプリケーションの設計または計画にタスクが関連することを示します。
  - 品質：製品の品質向上にタスクが関連することを示します。テスト、レビューの作成、自動テストの失敗の修正などが含まれます。
  - サポート：サポートにタスクが関連することを示します。
  - 管理：プロジェクト管理にタスクが関連することを示します。
  - 開発：アプリケーションの開発者にタスクが関連することを示します。これがデフォルトのタイプです。

- レビュー：レビューにタスクが関連することを示します。

#### カスタム タスク タイプの使用

カスタム値を入力してタスク タイプを表すこともできます。詳細については 277 ページの「カスタム フィールドの使用」を参照してください。

- 開始日：このタスクの作業を開始する日付を選択します。
- 予定時間：このタスクに対して作業する予定の時間を入力します。
- ストーリー ポイント：このタスクに関連付けるストーリー ポイントを入力します。詳細については 73 ページの「ストーリー ポイント」を参照してください。

---

## 新規タスクを作成するその他の方法

タスクを新規作成する方法はいくつかあります。

- 要求または欠陥 / 改善をスケジューリングします。106 ページの「要求と欠陥 / 改善のタスク化」を参照してください。
- [ 予定 ] ページで、ターゲットのイテレーションにバックログ項目をドラッグ アンド ドロップしてタスクを新規作成することもできます。詳細については 115 ページの「タスクとしてのバックログ項目のスケジューリング」を参照してください。

MS Project でタスクを作成することもできます。詳細については 207 ページの「Microsoft Project の統合」を参照してください。

## タスクの管理

[タスク] ページでは、プロジェクト中の自分のタスクを参照することができます。[タスク] ドロップダウンメニューの [表示] をクリックすると、[タスク] ページが開き、プロジェクトに属するすべてのイテレーションとタスクのリストが表示されます。以下の操作を実行できます。

- 新規タスクの作成 (123 ページの「タスクの新規作成」を参照)
- タスクのインポート (130 ページの「CSV からのタスクのインポート」を参照)
- タスクの一括更新 (125 ページの「タスクの一括更新」を参照)
- タスクのキャンセル (126 ページの「タスクのキャンセル」を参照)
- タスクの編集 (127 ページの「タスクの編集」を参照)





## タスクの一括更新







Project Center では、一度に複数のタスクを更新することができます。

1. 更新するタスクを選択し、[タスクの一括更新] をクリックします。

 Tasks

Search Details: Task Status: open; Task Project: Adamt1;

29 Item(s)    Quick New **Mass Update** 

<input type="checkbox"/>		ID	Name	Status
<input checked="" type="checkbox"/>		<a href="#">19</a>	User profile page - GUI front-end	open
<input checked="" type="checkbox"/>		<a href="#">21</a>	Shopping cart functionality - add, edit, removed items - GUI front end	open
<input checked="" type="checkbox"/>		<a href="#">23</a>	Coordination and planning activities	open
<input checked="" type="checkbox"/>		<a href="#">22</a>	Implement quick search	open
<input type="checkbox"/>		<a href="#">25</a>	Implement Req. 23 : Support for AmEx	open
<input type="checkbox"/>		<a href="#">24</a>	Test our app before passing to QA	open

2. [タスクの一括更新] ウィンドウでタスクの詳細を更新し (タスクのプロパティについては 123 ページの「タスクの詳細」を参照してください)、[保存] をクリックします。

### 成果物へのリンクの追加

Development Testing Platform の成果物（要求、タスク、欠陥、シナリオ、シナリオの実行、テストセッション）へのリンクを追加するには、次の書式を使用します。

- @task #
- @req #
- @pr #
- @scenario #
- @scenario run #
- @test session #

@ 記号は任意です。エンティティ名と ID の間に、空白以外の文字を入れてはいけません。

たとえば、次のコメントはタスク 123 にリンクします。

- This requirement should extend the functionality implemented in @task 123.

---

## タスクのキャンセル

いったん作成したタスクを完全に削除することはできません。ただし、作業範囲を妨げないよう、不要なタスクを変更できます。

1. [ タスク ] ページでキャンセルするタスクを選択します。

2. [一括更新] をクリックし、タスクのステータスを [キャンセル] に設定します。

Selected Tasks to update

4 Item(s)

ID	Name	Project	Status	Testing Required	Priority	Req.	Defect	Iter.	Owner(s)
<a href="#">19</a>	User profile page - GUI front-end	Adamt1	open	✓	Medium	<a href="#">4</a>		<a href="#">2</a>	admin
<a href="#">21</a>	Shopping cart functionality - add, edit, removed items - GUI front end	Adamt1	open	✓	Medium	<a href="#">11</a>		<a href="#">2</a>	DavidScott
<a href="#">22</a>	Implement quick search	Adamt1	open	✓	Medium	<a href="#">7</a>		<a href="#">2</a>	DavidScott
<a href="#">23</a>	Coordination and planning activities	Adamt1	open		Medium			<a href="#">2</a>	admin

Relation:

General

Project:  Iteration ID:   No Iteration

Priority:

Owner(s):  [\[Show\]](#) [\[Clear\]](#) Observer(s):  [\[Show\]](#) [\[Clear\]](#)

Status:  open  in progress  completed  canceled

Tags:

Testing Required:

Comment:

Links allowed in format: http|ftp|https://company.com, www.company.com, @task NNN, @req NNN (e.g. @task 111, @REQ 222)

3. [保存] をクリックします。

ステータスが [キャンセル] のタスクは、そのタスクが割り当てられているイテレーションでは表示されません。キャンセルしたタスクをさらに見えにくくするには、イテレーションへの割り当てを解除します。

## タスクの編集

タスク ID をクリックすると、タスクの詳細ページが開きます。

### 全般

[全般] タブでは、タスクのプロパティを変更できます。

- [ステータス] ドロップダウンメニューをクリックし、ステータスを変更します。
- タスクの完了に必要な見積り時間または実際の時間 (d, h, m 形式) を入力します。
- [新規コメント] セクションにコメントを入力します。
- [追加] をクリックし、画像などのタスクに関連する添付ファイルをアップロードします。
- [全般] タブでその他のタスクの詳細を編集する方法については、123 ページの「タスクの詳細」を参照してください。
- 編集が完了したら、[保存] をクリックします。

## サブタスクの管理

[サブタスク] タブをクリックすると、子タスクの参照と管理を行うことができます。



- 新しい子タスクを作成するには、[新規] をクリックします。詳細については 123 ページの「タスクの新規作成」を参照してください。
- タスク ID をクリックすると、詳細ページが開きます。



## 変更履歴の参照

[変更履歴] タブには、タスクに対して記録されたすべての変更が表示されます。次の情報が表示されます。

- **変更時刻**：タスクを変更した時刻。
- **ユーザー**：タスクを変更したユーザー。
- **変更されたフィールド**：変更したフィールド（たとえば、ステータス）。
- **削除された値**：変更したフィールドに関連する削除された値。
- **追加された値**：変更したフィールドに関連する追加された値。

 タスクの編集: [36] 要件 27 の実装: 音声認識

全般 | **変更履歴** | コード  506 | 自動テスト  0 | シナリオ  0

 **変更履歴** 

作成者: admin  
作成日時: 2010-10-13 17:25:41

変更されたフィールド	削除された値	追加された値
変更時刻: [2010-10-14 13:32:33] by admin		
見送り残作業時間	9d	5d 3h 58m
変更時刻: [2010-10-13 17:26:59] by admin		
ステータス	オープン	作業中
反復 ID		8.0
実作業時間	0d 0h 0m	2d 6h 30m
監督者		
見送り残作業時間	0d 0h 0m	9d
変更時刻: [2010-10-13 17:25:41] by admin		
オーナー		kylie
ステータス		オープン
テストが必要		true
プロジェクト		PolicyCdemo
予定作業時間		9d
優先度		中
名前		要件 27 の実装: 音声認識
要求 ID		27
開始予定日		2010-10-11
関係		要件の実装

 アクティビティ 

1 アイテム

ユーザー名	開始日	終了日	継続時間
kylie	2010-10-13 17:26		519d 6h

## タスクのソースコードの参照

[コード] をクリックすると、タスクに関連するコードを参照できます。

## 自動テスト

[自動テスト] タブをクリックすると、タスクに関連付けられた自動テストを参照できます。詳細については 204 ページの「自動テストの詳細」を参照してください。

## シナリオ

[シナリオ] タブをクリックすると、タスクに関連付けられたシナリオを参照および管理できます。テスト シナリオの作成および関連付けに関する詳細については 147 ページの「手動テスト シナリオの追加と編集」を参照してください。



## CSV からのタスクのインポート

Development Testing Platform で新規にタスクを作成するのに加えて、CSV ファイルからタスクをインポートすることができます。言い換えると、テキスト ファイルでタスクをあらかじめ定義するか、サードパーティ製のツールからタスクを CSV ファイルにエクスポートしておいて、後で Development Testing Platform にインポートすることができます。CSV ファイルからタスクをインポートするには、次の操作を行います。

- CSV のインポート設定
- インポート前のタスクのレビューと編集

**注意：** CSV ファイル中でタスク名などに各国語文字を使用している場合、Development Testing Platform にインポートする前に、必ず UTF-8 エンコーディングでファイルを保存してください。

## CSV のインポート設定

CSV ファイルから Development Testing Platform にタスクをインポートするには、次の操作を行います。

1. Project Center でツールバーの [CSV からタスクをインポート] アイコンをクリックします。



2. [タスクのインポート] ページが表示されます。

3. [ファイルを選択] ボタンをクリックし、インポートする CSV ファイルを指定します。
4. [プロジェクト] ドロップダウン リストから、タスクのインポート先になる Development Testing Platform プロジェクトを選択します。
5. (オプション) インポートしたタスクの作業を行うイテレーションの ID を指定します。
  - **注意:** イテレーション ID は CSV ファイルでも指定できます。ただし、[タスクのインポート] ページで定義されるイテレーション ID の方が優先されます。
6. [次へ] をクリックします。

ここまでの操作によって、続けてタスクのレビューと編集を行うことができます。

## インポート前のタスクのレビューと編集

オーナーや予定時間といったさまざまなタスク プロパティは、テキスト ファイルまたは CSV ファイルで定義できます。しかし、タスク名しかないタスクをインポートしてから [タスクのインポート] ページでタスク プロパティを変更または追加することもできます。タスクを削除することさえ可能です。

CSV のタスクをレビューまたは編集するには、次の操作を行います。

1. [次へ] をクリックすると、[タスクのインポート] ページが表示されます。



**注意：** CSV ファイルを作成する場合、1 行目はカラム名にし、カンマで区切ったすべてのデータを 2 行目以降に置くべきです。

2. 必要な変更を行ったら [保存] をクリックします。

インポートする前に、CSV ファイルがあるべき形式について確認するには、[ファイルフォーマット情報] リンクをクリックします。

CSV ファイルをアップロードし、ターゲットプロジェクトと反復を選択してください

ファイルパス:

インポートするタスクを含むファイルへのパス (csv 拡張子が必要) [\[ファイルフォーマット情報\]](#)

**注意：** [ 名前 ] カラムは必須ですが、他のカラムは任意です。

**CSV ファイル形式情報** [x]

CSV ファイルにタスクを定義し、後で Concerto にインポートできます。  
 先頭行にカラム ヘッダーを記述し、残りの行にデータを記述する必要があります。  
 カラム **[名前]** が必要です。その他は任意オプションです。詳細については、このページの **?** をクリックしてください。

**サンプル ファイルの例:**

```
名前
"Task #1 name"
"Task #2 name"
"Task #3 name"
"Task #4 name"
"Task #5 name"
"Task #6 name"
```

**注意:**  
 上記の名前のタスクが Concerto に作成されます。タスク ID は自動的に割り当てられます。  
 優先度およびタイプはデフォルト値 (「中」および「スタンド アロン」) になります。

**他のサンプル ファイル:**

```
名前,          オーナー, 予定
"Task #2 name",  "admin",  "5d"
"Task #3 name",  "admin",  "1h"
"Simple Task #4 name", "admin", "1h"
"Task #5 name",  "admin",  "2d"
"Task #6 name",  "admin",  "1d 1h"
```

Copyright (c) 1996-2012 Parasoft 閉じる

## 予測作業時間の変更

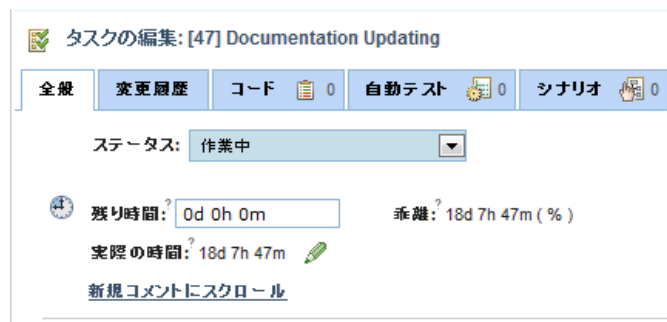
ユーザーがタスクに対して作業するとき、Development Testing Platform はタスクのステータスが変化しない時間を計測します。たとえば、タスクのステータスは「作業中」かもしれません（図 65）。計測された値は各タスクの「実際の時間」に表示されます。

図 65: タスクの検索結果 : 実際の時間



ID	オリジナル ID	名前	ステータス	優先度	反復	オーナー	予定	実際
47		Documentation Updating	作業中	中	7	nagata	0d 0h 0m	18d 7h 44m
98		FR 3 の実装: 改善要望	作業中	中	4	nagata	1d	6h 12m
100		PR 10 の修正: 表示の一部が欠けている	作業中	中	5	nagata	1d	24m
101		欠陥 11 の修正: 測定が安定しない	作業中	最高		nagata	1d	9m

図 66: [ タスクの編集 ] ページ : 実際の時間



タスクの編集: [47] Documentation Updating

全般 | 変更履歴 | コード 0 | 自動テスト 0 | シナリオ 0

ステータス: 作業中

残り時間: 0d 0h 0m | 乖離: 18d 7h 47m (%)

実際の時間: 18d 7h 47m

[新規コメントにスクロール](#)

「実時間」は自動的に計算されます。ただし、表示されている時間が実際の値と異なると判断した場合、ユーザーはタスクの「実時間」の値を変更できます。

たとえば、タスク オーナーが複数のタスクに対して作業した場合、またはタスクの完了後にタスクステータスを「処理中」から「完了」に変えるのを忘れた場合、表示される「実時間」の値は、実際にそのタスクにかかった時間よりも大きくなります。


タスクの「実時間」を変更するには、次の操作を行います。

1. [編集]アイコン  をクリックします。

**注意:** 入力フィールドに表示される文字列を修正して変更することができます。変更を中断するには Esc キーを押します。

2. 必要に応じて「実時間」を変更します。
3. [保存] ボタンをクリックします。

変更した値が保存されます。

タスクの「実時間」を変更すると、黄色の警告アイコン  が表示されます。このアイコンのツールチップには、追加 / 削減された時間が表示されます。通常のタスクの変更と同様に、関係者にタスクの変更についての電子メールが送信されます。

## タスクの検索

Project Center では、タスクを作成し、タスクとして成果物をスケジュールし、タスクを成果物と関連付けます。これらのタスクはすべて [タスク検索] ページを使って探すことができます。

タスクを検索するには、次の操作を行います。

1. Project Center で [タスク] メニューの [検索] をクリックします。[タスク検索] ページが表示されます。

図 4: [タスク検索] ページ

全般

タスク ID: <input type="text"/>	タスク ステータス: <input type="text" value="オープン"/>
オリジナル ID: <input type="text"/>	<input type="text" value="作業中"/>
タスク名: <input type="text"/> <input type="checkbox"/> Not	タスク優先度: <input type="text" value="最高"/>
説明/コメント: <input type="text"/>	<input type="text" value="高"/>
	<input type="text" value="中"/>

タスクの関係

反復 ID: <input type="text"/> <input type="button" value="検索"/>	タスク プロジェクト: <input type="text" value="PolicyCdemo"/>
反復内にある: <input type="text" value="任意"/>	<input type="text" value="ReportCenterTest_Project"/>
要求 ID: <input type="text"/> <input type="button" value="検索"/>	<input type="text" value="Shapes"/>
欠陥 ID: <input type="text"/> <input type="button" value="検索"/>	<input type="text" value="SOAtestプロジェクト"/>
タグ: <input type="text"/> <input type="checkbox"/> Not	タスク オーナー: <input type="text"/> <input type="checkbox"/> 空
	作成者: <input type="text"/>

タスクテスト

テスト ステータス: <input checkbox"="" type="text" value="---&lt;/input&gt;&lt;/td&gt; &lt;td&gt;(再)テストが必要: &lt;input type="/>	
タイプ: <input type="text" value="--制約なし--"/>	

2. 次の 1 つ以上の検索条件を入力します。

- **ID、名前、オーナー、ステータス、優先度、説明** 一般的なタスク プロパティ。
- **[エンティティ] ID** タスクが関連付けられる可能性がある他のプロジェクト成果物 (要求、イテレーション、欠陥 / 改善) の ID。
- **タスク プロジェクト** 検索されるタスクが属する Development Testing Platform プロジェクト。
- **テスト ステータス** 検索されるタスクに関連付けられたテストのステータス。
- **(再)テストが必要** 検索されるタスクにテストまたは再テストが必要かどうか。
- **タスクの作成 / 変更** 検索されるタスクが作成 / 変更された可能性がある期間。

3. [検索] または [Search & Hide Criteria] ボタンをクリックします。

検索結果が [ タスクの検索 ] の表に表示されます。

図 5: 検索結果

タスク							
検索の詳細: タスクプロジェクト: PolicyCdemo;						保存済み検索: -カス	
53 アイテム <span>一括更新</span>							
<input type="checkbox"/>		ID	名前	ステータス	テスト	優先度	反復
<input type="checkbox"/>		<a href="#">35</a>	要件 25 の実装: 本体の組み上げ	完了	成功	高	<a href="#">8</a>
<input type="checkbox"/>		<a href="#">36</a>	要件 27 の実装: 音声認識	作業中	未テスト	中	<a href="#">8</a>
<input type="checkbox"/>		<a href="#">37</a>	要件 28 の実装: 命令の実行(走る、持つ、座る)	作業中	未テスト	中	<a href="#">8</a>
<input type="checkbox"/>		<a href="#">106</a>	Architecture design - top level view	完了	不要	高	<a href="#">29</a>
<input type="checkbox"/>		<a href="#">107</a>	Database structure design	完了	不要	中	<a href="#">29</a>
<input type="checkbox"/>		<a href="#">108</a>	Easy user registration in the shop - server side	完了	成功	中	<a href="#">29</a>
<input type="checkbox"/>		<a href="#">109</a>	Easy user registration in the shop - GUI front-end	完了	未テスト	中	<a href="#">29</a>
<input type="checkbox"/>		<a href="#">110</a>	Forgot password functionality	完了	失敗	中	<a href="#">29</a>
<input type="checkbox"/>		<a href="#">111</a>	Fix 欠陥 51 : National characters do not work in user profile data	完了	不要	中	<a href="#">29</a>
<input type="checkbox"/>		<a href="#">112</a>	User profile page - server side	完了	成功	中	<a href="#">29</a>

[ タスクの検索 ] の表には次の情報が表示されます。

- **タスク ID, 名前, ステータス, 優先度** 基本的なタスク プロパティ。
- **関係** 要求を実装するためのタスクである場合、要求 ID が表示されます。欠陥を修正するためのタスクである場合、欠陥 ID が表示されます。
- **イテレーション** タスクが属するイテレーション。
- **オーナー** タスクを担当するユーザー。1 つのタスクが複数のユーザーに割り当てられる場合もあります。
- **予定** このタスクを完了するための予定された日数。
- **残り** このタスクを完了するための残り時間。

## イテレーションの編集

[イテレーションの編集] ページ ( 図 7) には、一般的な情報、作業時間の詳細、関連する添付ファイルが表示されるほか、表示されているイテレーションを完了するために必要なタスクの一覧が表示されます。

イテレーションとタスクの管理方法の詳細については、141 ページの「イテレーション内のタスクの管理」を参照してください。

[イテレーションの編集] ページを表示するには、次の操作を行います。

1. [イテレーション] ドロップダウンメニューの [検索] を選択します。
2. 適切な検索条件を入力して [検索] ボタンをクリックします。

全般

3. 入力した検索条件に基づいて検索結果が表示されます。

図 6: 検索結果

ID	名前	開始日	終了日
<a href="#">8</a>	フェーズ1	2010-10-08	2010-10-20
<a href="#">29</a>	Milestone 1: Architecture design and core framework implementation	2012-07-26	2012-08-25
<a href="#">30</a>	Milestone 2: Basic functionality implementation	2012-08-26	2012-09-25
<a href="#">31</a>	Milestone 3: Advanced functionality implementation and release	2012-09-26	2012-10-26

4. 編集するイテレーションの ID をクリックします。

[イテレーションの編集] ページ ( 図 7) が表示されます。



図 7: [イテレーションの編集] ページ

反復の編集: [30] Milestone 2: Basic functionality implementation

全般 タスク 12 タスク オーナー別の作業 バーンダウン チャート 統計

全般

プロジェクト: PolicyCdemo

名前: Milestone 2: Basic functionality implementation

説明: In this iteration we should implement all the basic features, so that our Online Shop is functional and can be used for demonstration purposes (not in production yet)

スケジュール済みリソース

Dev.#: 3.0 66d

作業時間

予定

開始日: 2012-08-26 終了日: 2012-09-25

予定時間: 54d 保存 削除 MSプロジェクトで開く

[イテレーションの編集] ページには次の情報が表示されます。

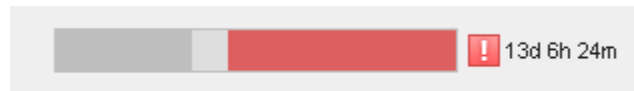
#### 全般

- **名前:** イテレーションの名前。
- **説明:** このイテレーションについての説明。
- **プロジェクト:** このイテレーションが属するプロジェクトの名前。

#### スケジュール済みリソース

- **Dev.#:** このイテレーションでの作業をスケジュールされたチームメンバーの数。チームメンバーが他の作業と掛け持ちになる場合、たとえば「.5」のように小数で表すことができます。
- **イテレーション キャパシティ バー:** イテレーションのキャパシティおよびスケジュールされたリソースをグラフィカルに表します。キャパシティは、予定作業時間（イテレーションの開始から終了まで）およびこのイテレーションに割り当てられたチームメンバーの数によって自動的に計算されます。イテレーション キャパシティ バーは以下のことを表します。
  - すでに何日の作業日が経過したか。
  - 今後のタスクとして何日の作業日がスケジュールされているか。
  - まだスケジュールされていない作業日は何日あるか。
  - 過剰にスケジュールされた作業日、つまり実際のイテレーションのキャパシティよりも多くのタスクがスケジュールされている作業日が何日であるか。つまり、

図 8: イテレーション キャパシティ バー



### 作業時間

- **開始日**：このイテレーションの作業を開始する予定の日。
- **終了日**：このイテレーションの作業を終了する予定の日。
- **予定時間**：このイテレーションに予定された作業時間の合計。このイテレーションに含まれるすべてのタスクの「予定時間」の合計です。
- **実際の時間**：このイテレーションの作業にかかった時間。このイテレーションに含まれるすべてのタスクの「実際の作業時間」の合計です。
- **残り時間**：このイテレーションが完了するまでに残っている作業時間の見積り。このイテレーションに含まれるすべてのタスクの「残り時間」の合計です。
- **終了日**：Project Center がタスクが完了すると予測した日。この日付は、最も作業量の多いプロジェクト タスクを担当しているチーム メンバーが、最期のタスクを完了する見込みの日付です。

### 添付

- **名前**：ファイル名。[ 追加 ] をクリックして添付ファイルをアップロードします。
- **説明**：添付ファイルについての説明。
- このページを使って、イテレーション中のタスクを管理し、終了日までに完了するようにタスクをチーム メンバー間で調整して配分することもできます。詳細については 141 ページの「イテレーション内のタスクの管理」を参照してください。

### タスク

- **[ タスク ] タブ**：このイテレーションを終了するために完了しなければならないタスクがすべて表示されます。このリストはソートすることができ、ステータス、プロパティ、オーナーといった情報を提供します。31 ページの「タスクのクイック変更」の説明にあるように、タスクの表から直接タスクを変更することができます。
- **[ タスク オーナー別の作業 ] タブ**：以下の情報が表示されます。各チーム メンバーの作業量を見ることができます。
  - **オーナー**：チーム メンバーのユーザー名。
  - **予定作業時間合計**：このイテレーションを担当するチーム メンバーに割り当てられた作業時間。このイテレーションを担当するチーム メンバーに割り当てられたすべてのタスクの「予定作業時間」の合計。
  - **実作業時間合計**：このイテレーションを担当するチーム メンバーがすでに使用した作業時間。このイテレーションを担当するチーム メンバーに割り当てられたすべてのタスクの「実時間」の合計。
  - **見積り残作業時間合計**：このイテレーションを担当するチーム メンバーがイテレーションを完了するための残作業時間の予測。このイテレーションを担当するチーム メンバーに割り当てられたすべてのタスクの「残作業時間」の合計。

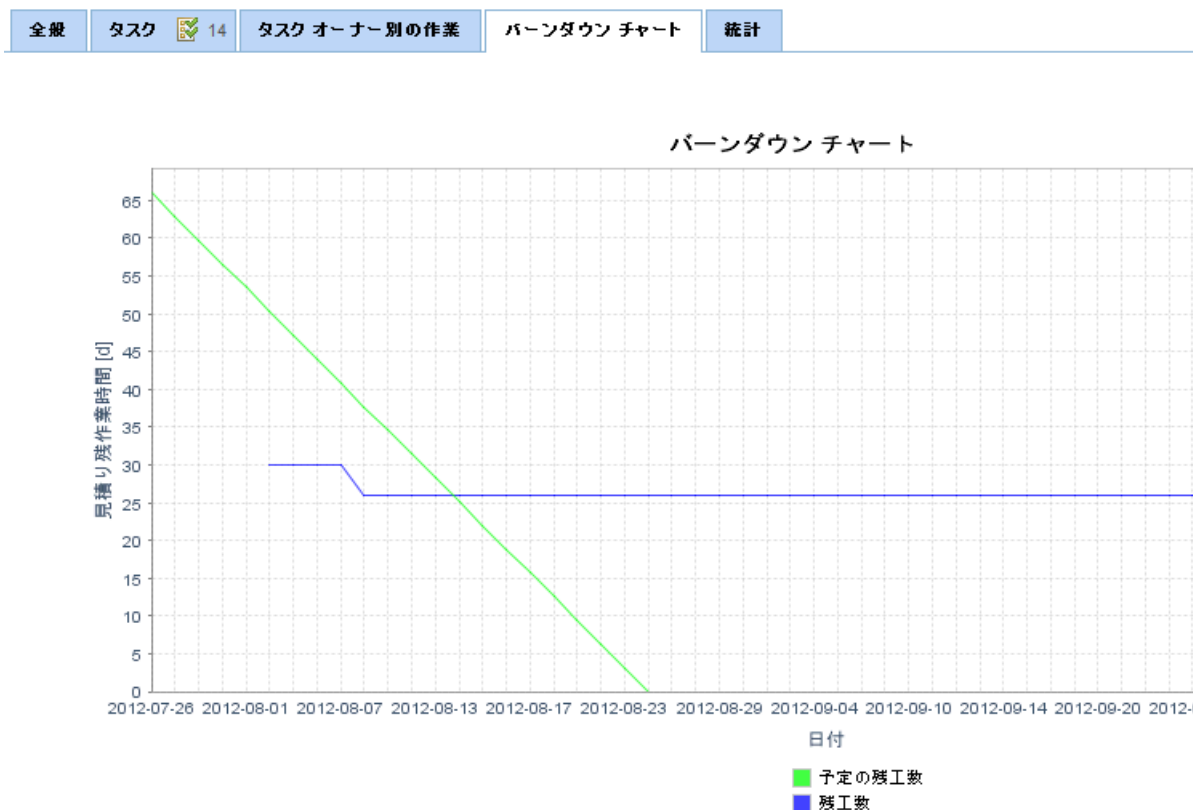
- **[バーンダウン チャート] タブ**：イテレーションにおける理想的な作業進捗と実際の作業進捗の傾向を表示します。バーンダウン チャートを使用すると、最終期限までにイテレーションを確実に完了させるために、作業の進捗をトラッキングすることができます。

バーンダウン チャートは 2 本の線から構成されます。

- **緑**：イテレーションにおける作業進捗の理想的な傾向。  
 緑の線は、イテレーションの開始日に始まり、イテレーションの終了日で終わります。
- **青**：イテレーションにおける作業進捗の実際の傾向。  
 青の線は、イテレーション中の最初のタスクが作成された日に始まります（必ずしも開始日には一致しません）。青の線は次のいずれかで終わります。
  - バーンダウン チャートがアクセスされた日。
  - イテレーションが終了した日（文字どおり、残作業時間の値に対する最終変更日）。

図 9 は、2 つのイテレーションにおける作業進捗の傾向を表したものです。

図 9: バーンダウン チャート: 実際の傾向



実際の傾向は、作業負荷が減ったことを示すほか、イテレーションの終了日に間に合わせるには、さらに加速して減り続けなければならないことを示します。

# イテレーション内のタスクの管理

イテレーションのコンテキスト内でタスクの参照と管理を行うことができるため、チームメンバーの作業バランスを調整するのが容易です。効果的にイテレーションの終了日の予測と評価を行い、各タスクの開始日を設定することができます。

1. Project Center のメインメニューで [ イテレーション ] タブをクリックします。
1. イテレーションの ID をクリックすると、[ イテレーションの編集 ] ページが開きます。

ID	Name	Start Date	End Date	Total S.P.	Remaining S.P.
2	Milestone 1: Architecture design and core framework implementation	2013-08-16	2013-09-15	0.0	0.0
3	Milestone 2: Basic functionality implementation	2013-09-16	2013-10-16	0.0	0.0
4	Milestone 3: Advanced functionality implementation and release	2013-10-17	2013-11-16	0.0	0.0

[ イテレーションの編集 ] ページでは、イテレーションの名前やイテレーションが関連付けられているプロジェクト等を変更できます。詳細については 137 ページの「イテレーションの編集」を参照してください。

反復の編集: [30] Milestone 2: Basic functionality implementation

全般 タスク 12 タスク オーナー別の作業 バーンダウン チャート 統計

全般

プロジェクト: PolicyCdemo

名前: Milestone 2: Basic functionality implementation

説明: In this iteration we should implement all the basic features, so that our Online Shop is functional and can be used for demonstration purposes (not in production yet)

スケジュール済みリソース

Dev.#: 3.0

作業時間

予定

開始日: 2012-08-26 終了日: 2012-09-25

予定時間: 54d

保存 削除 MSプロジェクトで開く

以下のセクションでは、イテレーションのコンテキスト内でタスクを管理する方法について説明します。Project Center の「一括更新」機能を使ってプロパティを共有する複数のタスクを一度に管理することもできます。タスクの一括更新については 125 ページの「タスクの一括更新」を参照してください。


## 作業配分の評価

作業配分を評価するには、次の操作を行います。


1. [ タスク オーナー別の作業 ] タブをクリックします。
2. [ 予定作業時間合計 ] のヘッダーをクリックしてソートします。

図 10:[ イテレーションの編集 ]- タスク オーナー別の作業

反復の編集: [29] Milestone 1: Architecture design and core framework implementation

全般 タスク  14 タスク オーナー別の作業 **バーンダウン チャート** 統計

4アイテム

オーナー ↕	予定作業時間合計 ▼	実作業時間合計 ↕	見残り残作業時間
nagata			
admin	13d	1d	8d
DavidScott	15d		 10d
JohnSmith	26d		8d

作業量を均等にするために、あるオーナーのタスクの一部を別のオーナーに割り当てることができます。下記の「別のユーザーへのタスクの再割り当て」を参照してください。

## 別のユーザーへのタスクの再割り当て

あるユーザーから別のユーザーにタスクを再割り当てする目的は、作業を均等に配分することです。その際、Project Center は「終了日」を再計算します。通常、計算後の日付の方が近い日付になります。

あるユーザーから別のユーザーにタスクを再割り当てするには、タスクのオーナーを変更します。

1. 図 10 にあるように、[ イテレーションの編集 ] ページに [ タスク オーナー別の作業 ] タブが表示されているものとします。[ タスク ] タブをクリックします。





2. 再割り当てするタスクの  [ クイック変更 ] をクリックします。

図 11 にあるように、このタスクがハイライト表示されています。

図 11: [イテレーションの編集]-タスク

全般		タスク  14	タスク オーナー別の作業	バーンダウン チャート	統計			
クイック新規タスク追加								
名前:		<input type="text"/>	オーナー: <input type="text" value="ESU"/>	<a href="#">表示</a> <a href="#">削除</a>	予定作業時間: <input type="text"/> d <input type="text"/> h <input type="button" value="作成"/>			
14 アイテム <span style="float:right">▼</span>								
<input type="button" value="新規タスク"/> <input type="button" value="既存のタスクをインポート"/> <input type="button" value="一括更新"/>								
<input type="checkbox"/>	ID	名前	ステータス	テスト	優先度	反復	オーナー	予定
<input type="checkbox"/>	 106	Architecture design - top level view	完了	不要	高	29	admin	5d
<input type="checkbox"/>	 107	Database structure design	完了	不要	中	29	JohnSmith	5d
<input type="checkbox"/>	 108	Easy user registration in the shop - server side	完了	成功	中	29	JohnSmith	4d
<input type="checkbox"/>	  109	Easy user registration in the shop - GUI front-end	完了	未テスト	中	29	JohnSmith	5d
<input type="checkbox"/>	 110	Forgot password functionality	完了	失敗	中	29	JohnSmith	4d

- [オーナー] フィールドで、タスクを再割り当てるオーナーのユーザー名を入力します。複数のオーナーのユーザー名を入力できます。
- 次のいずれかの操作を行います。
  -  [変更をコミット] をクリックして変更を確定します。

-  [変更をキャンセル] をクリックして変更をキャンセルします。

これでタスクが再割り当てされます。

- 作業量が変わったことを確認するために、[タスク オーナー別の作業] タブをクリックします。

Project Center が自動的に「予定作業時間合計」を再計算します。タスクは前よりも均等に配分されています。

Project Center は「終了日」も自動的に再計算します。タスクの再割り当てによって、そのオーナーの「終了日」が変化します。

## タスクの作業順序

すべてのタスクをチームメンバーに割り当てたら、タスクを実行する順序を決定します。次の方法があります。

- タスクの優先順位の変更
- 開始日と予定時間の変更

## タスクの優先順位の変更

タスクの優先順位を変更し、優先順位が最も高いタスクから作業を開始するよう、チームに通知することができます。

タスクの優先順位を変更するには、次の操作を行います。

- [タスク] タブをクリックします。

2. 再割り当てするタスクの隣の  [クイック変更] をクリックします。

図 11 にあるように、このタスクがハイライト表示されます。



3. 変更するタスクの行で [優先度] フィールドをクリックします。

ドロップダウン リストが表示されます。



4. タスクに割り当てる優先順位を選択します。

5. 次のいずれかの操作を行います。

-  [変更をコミット] をクリックして変更を確定します。
-  [変更をキャンセル] をクリックして変更をキャンセルします。

タスクの優先順位が更新されます。

## 開始日と予定時間の変更

タスクの開始日を変更して、チームメンバーが作業するタスクの順序を直接指定することができます。

タスクの開始日を変更するには、次の操作を行います。

1. 変更するタスクのタスク ID をクリックします。

[タスクの編集] ページが表示されます。

2. [開始日] フィールドで、カレンダーから適切な日付を選択します (図 12)。

図 12: タスクの編集 - 開始日

タスクの編集: [113] User profile page - GUI front-end

全般 | 変更履歴 | コード 0 | 自動テスト 0 | シナリオ 0

ステータス: 作業中

残り時間:  垂離:

実際の時間:

新規コメントにスクロール

要求の実装 94 | 説明およびコメント (1) | 添付 (0)

説明およびコメント

説明

Implement web pages for user profile page.  
For GUI please consult graphics designers team.

新規コメント:

添付

0 アイテム | 追加

名前	説明	添付日

プロジェクト: PolicyCdemo  
名前: User profile page - GUI front-end  
反復 ID: 29 | 検索 | 追加 | ヒュ

優先度: 中

オーナー: admin | 表示 | 閉  
監督者: | 表示 | 閉  
作成者: admin

タイ  
テストが必  
タ

October, 2012						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
40	30	1	2	3	4	5
41	7	8	9	10	11	12
42	14	15	16	17	18	19
43	21	22	23	24	25	26
44	28	29	30	31	1	2
45	4	5	6	7	8	9

Today

3. [ 予定時間 ] フィールドで、タスクを完了する予定の日を入力します。
  4. [ 保存 ] をクリックします。
  5. [ イテレーションの編集 ] ページの [ タスク ] タブに戻って [ リフレッシュ ] をクリックします。
- タスク リストの [ 予定 ] カラムに予定作業時間が表示されます ( 図 13)。

図 13: タスク - 予定作業時間

ID	名前	ステータス	テスト	優先度	反復	オーナー	予定
113	User profile page - GUI front-end	作業中	未テスト	中	29	admin	5d
114	Shopping cart functionality - add, edit, removed items - server side functionality	作業中	未テスト	中	29	JohnSmith	5d
115	Shopping cart functionality - add, edit, removed items - GUI front end	オープン	未テスト	中	29	DavidScott	6d



# Project Center でのテスト

このセクションの内容：

- 手動テスト シナリオの追加と編集
- テスト シナリオのインポート
- 受け入れテスト セッションの実行
- Project Center からの SOAtest の実行

---

## 手動テスト シナリオの追加と編集

シナリオに記述された機能を実装したら、手動テスト シナリオを作成して実行し、機能が正しく実装されたことを検証する必要があります。

チーム メンバーは段階的なテスト シナリオを定義し、関連するタスクに割り当てることができます。手動テスト レポートから、個々の手動テスト シナリオについて、成功したもの、失敗したもの、テストの完了にかかった時間といった詳細情報を参照できます。キャプチャされる個々の手動ステップを利用して、完全な手動シーケンスを簡単に取得、レビュー、および繰り返すことができます。

企業のポリシーに応じて、テスト シナリオは次のように収集および作成されます。

- 要求の指定中に収集されて作成される（要求に割り当てられる）。
- 開発中に収集されて作成される。
- 開発が完了した後に収集されて作成される。

158 ページの「受け入れテスト セッションの実行」の説明にあるように、手動テスト シナリオ セットは受け入れテストの手動テスト セッションのために実行することができます。

特定の要求、欠陥 / 改善、タスクに関連があってもなくても、手動テストは追加できます。

---

## 要求、欠陥 / 改善、タスクとテスト シナリオとの関連付け

相関を指定する方法はいくつかあります。

- 相互に関連付けられたシナリオの新規作成
- 既存のテスト シナリオと要求 / 欠陥との関連付け
- シナリオ実行と欠陥との関連付け

---

### 相互に関連付けられたシナリオの新規作成

特定の要求、欠陥 / 改善、タスクを検証するための新しいテスト シナリオを追加するには、次の操作を行います。

1. テスト シナリオを追加する要求、欠陥 / 改善、またはタスクを開きます。

- **注意:** 次の図は要求の編集画面ですが、欠陥 / 改善とタスクでもプロセスは同じです。

図 14: 要求の編集

🔗 要求の編集: [93] Easy user registration in the shop

全般 コラボレーション タスク 4 欠陥 0 変更履歴 コード 0 自動テスト 0 シナリオ 1/1

📄 全般

プロジェクト: PolicyCdemo

名前: Easy user registration in the shop

説明: it should be comfortable and almost seamless for the user to register in our shop. Let's require only the most basic data like email. The geographical location should be drawn from internet traffic data.

ランク: 50000

オリジナル ID:

ステータス: 作業中

期限:

オーナー:

🔍 作業量

ストーリー ポイント: 子ストーリー ポイント: 0.0 合計: 0.0

👥 関係

親要求 ID: 91 [検索] [ビュー]

🔗 子要求 [新規] [割り当て]

📄 詳細 [保存]

2. [シナリオ] タブをクリックします。

図 15: [シナリオ] タブ

🔗 要求の編集: [93] Easy user registration in the shop

全般 コラボレーション タスク 4 欠陥 0 変更履歴 コード 0 自動テスト 0 シナリオ 1/1

1 アイテム [新規] [割り当て] [割り当て解除]

ID	名前	シナリオ実行結果	実行者	テスト履歴	実行
20	User registration in the shop - common scenario	2012-08-02 15:18	admin		

3. [新規] をクリックします。
4. 表示されたページで、プロジェクトを選択し、新しいシナリオの名前と説明を入力し、ステップを定義します。そして [作成] をクリックします。

ここまでの操作によって、158 ページの「受け入れテスト セッションの実行」で説明しているように、テストセッションにテストをスケジューリングして実行できるようになりました。

---

## 既存のテスト シナリオと要求 / 欠陥との関連付け

71 ページの「欠陥、改善、テスト、タスクの関連付け」を参照してください。

---

## シナリオ実行と欠陥との関連付け

72 ページの「欠陥とシナリオ実行の関連付け」を参照してください。

---

---

## 関連付けられていないテストの追加

特定のタスク、要求、欠陥 / 改善に関連付けられていない手動テスト シナリオを追加するには、次の操作を行います。

1. [ テスト ] ドロップダウン メニューの [ 新規 ] をクリックします。
2. 表示されたページで、プロジェクトを選択し、新しいシナリオの名前と説明を入力し、ステップを定義します。そして [ 作成 ] をクリックします。
3. [ 保存 ] をクリックしてテスト シナリオを保存します。

ここまでの操作によって、158 ページの「受け入れテスト セッションの実行」で説明しているように、テスト セッションにテストをスケジューリングして実行できるようになりました。

---

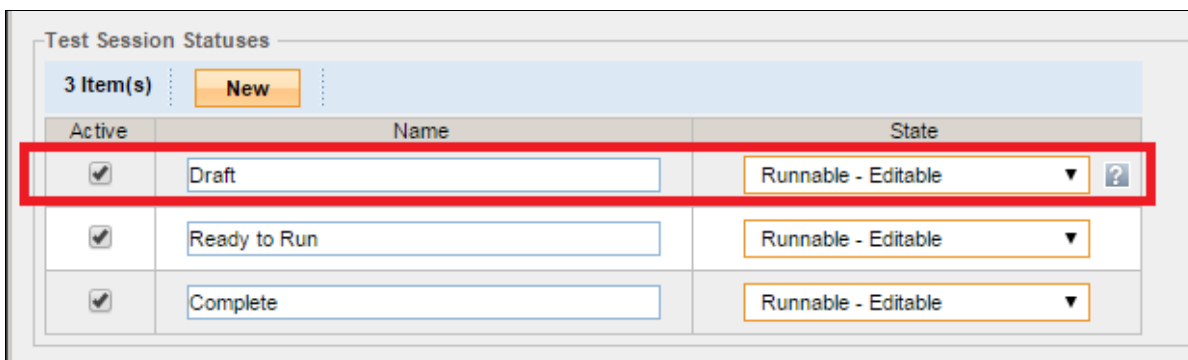
## テスト セッションの複製

1. テスト セッションを開き、[ 複製 ] をクリックします。
  2. セッションに関連付けるシナリオを選択します。
  3. [ 複製 ] をクリックします。
- 

## 複製されるテスト セッションのデフォルト ステータスの設定

Project Center の管理ページで [ 重複 ] オプションを使って作成した新規のテスト セッションについて、そのステータスをカスタマイズすることができます。

[ 管理 ] リンクをクリックして Project Center の管理ページを開き、[ 設定 ] > [ 成果物のステータス ] を選択します。[ テスト セッションのステータス ] エリアで 1 番目にアクティブな項目が、新規に複製されるテスト セッションのステータスとして使用されます。



## 既存のテスト シナリオを開く

既存のシナリオ（名前、ステップなど）を開く方法は主に 2 つあります。

- [ テスト ] > [ 検索 ] や他の検索ツールを使ってテストを探し、シナリオ ID をクリックします。



- 関連する要求、タスク、または欠陥 / 改善のページを開いて [ シナリオ ] タブに移動し、シナリオの ID をクリックします。

ここから、以下の作業を行うことができます。

[ シナリオ ] タブ：一般的なシナリオの詳細を編集するだけでなく、テスト ステップの編集、挿入 / 削除、および並べ替えを行います。

[ 添付 ] タブ：添付ファイルの参照、追加、および削除を行います。

[ テスト履歴 ] タブ：シナリオのテスト実行履歴を参照します。

[ 変更履歴 ] タブ：シナリオの変更履歴を参照します。

## テスト シナリオへの添付ファイルの追加

既存のテスト シナリオに添付ファイル（画面ショット、Excel ファイル、Word ファイルなど）を追加するには、次の操作を行います。

1. 添付ファイルを追加するシナリオの [ シナリオの編集 ] ページを開きます。
2. [ 添付 ] タブをクリックします。
3. [ 追加 ] をクリックして添付ファイルを追加します。

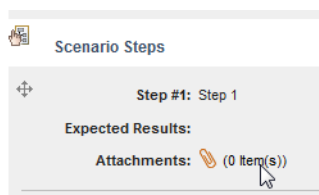
#### MySQL データベースのデフォルトの最大パケット サイズは 1MB です

このデフォルト設定のために、添付ファイルの最大ファイル サイズは 1MB に制限されます。MySQL データベースの最大パケット サイズを変更せずに 1 MB より大きいファイルを添付しようとすると、エラーが発生する可能性があります。

## 特定のテスト ステップへの添付ファイルの追加

既存のテスト シナリオ中の特定のテスト ステップに添付ファイル ( 画面ショット、Excel ファイル、Word ファイルなど ) を追加するには、次の操作を行います。

1. 添付ファイルを追加するシナリオの [ シナリオの編集 ] ページを開きます。
2. [ シナリオ ] タブをクリックし、添付ファイルを追加するテスト ステップの [ 添付 ] アイコンをクリックします。



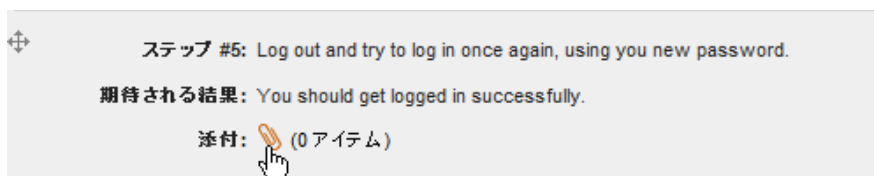
新規に添付ファイルを追加するためのコントロールが表示されます。

3. コントロールを使用して添付ファイルを追加します。

## テスト ステップの並べ替え

テスト ステップを並べ替えるには、次の操作を行います。

1. テスト ステップを並べ替えるシナリオの [ シナリオの編集 ] ページを開きます。
2. [ シナリオ ] タブで、各ステップの左側にあるアイコンをドラッグ アンド ドロップしてテスト ステップの順番を変更します。



## シナリオの一括更新

[シナリオの管理] ページなどで、シナリオのリストから複数のシナリオを一度に更新するには、次の操作を行います。

1. 更新するシナリオのチェックボックスをオンにします。
2. [一括更新] をクリックします。
3. 変更を行って [保存] をクリックします。

## シナリオの削除

[シナリオの管理] ページのリストからシナリオを削除する方法は 2 つあります。

- 削除するシナリオをクリックし、[選択されたアイテムの削除] アイコンをクリックします。
- シナリオを右クリックし、[削除] をクリックします。

### 削除できない場合

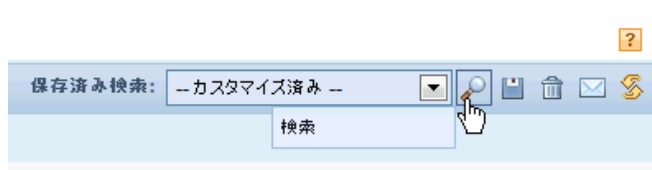
Development Teting Platform を制限モードで設定している場合、成果物を削除することはできません (詳細については『管理者ガイド』の 321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。

使用しない成果物を削除するのではなく、現在のプロジェクトから専用のプロジェクトに (たとえば Obsolete という名前のプロジェクトに) 成果物を移動できます。

## 削除したシナリオの復元

過去に削除したシナリオを復元する必要がある場合、次の操作を行います。

1. 次の操作を行って、削除したシナリオを探します。
  - a. [テスト] ドロップダウン メニューの [管理] をクリックします。
  - b. ページ右上の [検索] アイコンをクリックします。



- c. 表示されたダイアログで [ 削除済 ] チェックボックスをオンにします。

シナリオ条件

ID:

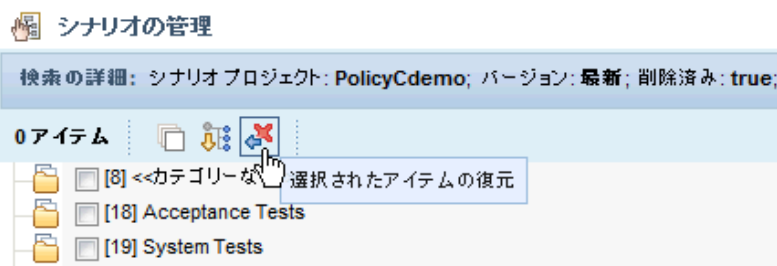
名前:

説明:

プロジェクト: PolicyCdemo

削除済み:

- d. (オプション) さらに検索条件を追加します。
- e. [ 検索 ] をクリックします。
2. 復元するシナリオのチェックボックスをオンにします。
  3. [ 選択されたアイテムの復元 ] ボタンをクリックします。





---

# テスト シナリオのインポート

テスト シナリオは、csv、xls、および xlsx ファイルからインポートすることができます。HP Quality Center などのテスト プラットフォームから Excel や csv ファイルとしてテストをエクスポートし、それを Development Testing Platform にインポートすることができます。

csv ファイル中でシナリオの名前などに National 文字を使用している場合、Development Testing Platform にインポートする前に、必ず UTF-8 エンコーディングでファイルを保存してください。

ファイルからテスト シナリオをインポートするには、次の操作を行います。

1. [テスト] ドロップダウン メニューの [インポート] をクリックします (またはテスト ページで [インポート] アイコンをクリックします)。
2. [ステップ 1: ターゲット プロジェクトの選択] セクションで、テスト シナリオが属するプロジェクトを指定します。
3. [ステップ 2: ファイルの選択 (.csv, .xls, .xlsx)] セクションで [ファイルの選択] ボタンをクリックします。ファイル選択ダイアログで適切なファイルを選択して [開く] ボタンをクリックします。
4. [ステップ 3: マッピングの構成] セクションで、ファイル中の異なるカラムを Development Testing Platform のシナリオ フィールドにマッピングします。必須なのは [名前] フィールドだけです。[ステップ番号] と [ステップの説明] の指定はどちらも任意です。ただし、[ステップ番号] がインポートされる場合、[ステップの説明] も必要です。
  - [ステップ番号] は、シナリオでのステップの位置を指す数字です。[ステップ番号] は 1 から開始し、1 ずつ増分します。
  - [ステップの説明] は、ステップを実行するための説明です。[ステップ番号] を設定する場合、[ステップの説明] も必要です。

 ファイルからシナリオをインポート

ステップ 2: ファイルの選択 (.csv, .xls, .xlsx) [\[ファイルフォーマット情報\]](#)

ファイルパス: \*

+ ファイルの選択

検索シナリオテスト.xlsx

終了

ステップ 3: マッピングの構成

インポートされたフィールド *	シナリオのフィールド	選択されたファイルからのカラム名
	名前	シナリオ名
	説明	シナリオの説明
	ステップ番号	ステップ番号
	ステップの説明	ステップの説明
	期待される結果	---
	ステータス	---

### カスタム フィールドのインポート

277 ページの「カスタム フィールドの使用」で説明しているように、管理者によって追加されたカスタム フィールドをインポートしたりマッピングすることができます。

5. [次へ] をクリックします。
6. 提案されたマッピングをレビューします。
7. 提案されたマッピングで問題ない場合、[Submit] をクリックします。問題がある場合、[戻る] をクリックして修正します。

オプションを設定して、インポートされるシナリオと既存のシナリオが重複する場合に、新しいシナリオで既存のシナリオを置き換えるか、それともインポートを止めるかを指定します。

## ファイル フォーマット

インポートする Excel ファイルは以下の形式に従うべきです。

Scenario Name	Scenario Description	Step Number	Step Description
Scenario 1	Scenario 1 Description	1	Step 1 Description
Scenario 1	[ 空 ]	2	Step 2 Description
Scenario 2	Scenario 2 Description	[ 空 ]	[ 空 ]

- 1 行目は、上記のように 4 個のカラム名でなければなりません。2 行目以降の行はそれぞれ 1 つのシナリオ ステップを扱うべきです。
- シナリオごとに 1 つの説明を用意するべきです。
- 1 番目のステップに結び付ける [Scenario Description] を一度用意する必要があります。その後、この [Scenario Description] はシナリオ全体に適用されます。2 回以上 [Scenario Description] を記述することができます。
- 1 つのタスクのすべてのステップは、同じファイル中で新しいシナリオを開始する前に置かなければなりません。たとえば、シナリオ 1 は 5 ステップあり、シナリオ 2 は 3 ステップあるものとします。
  - 許容されない：
    - Scenario 1: step 1
    - Scenario 1: step 2
    - Scenario 2: step 1
    - Scenario 2: step 2
    - Scenario 2: step 3
    - Scenario 1: step 3
    - Scenario 1: step 4
    - Scenario 1: step 5
  - 許容される：
    - Scenario 1: step 1
    - Scenario 1: step 2
    - Scenario 1: step 3
    - Scenario 1: step 4
    - Scenario 1: step 5
    - Scenario 2: step 1
    - Scenario 2: step 2
    - Scenario 2: step 3

- [Step Number] および [Step Description] の指定はどちらも任意です。ただし、[Step Number] がインポートされる場合、[Step Description] も必要です。
  - [Step Number] は、シナリオでのステップの位置を指す番号です。[Step Number] は 1 で開始し、1 ずつ増分します。
  - [Step Description] は、ステップを実行するための説明です。[Step Number] を設定する場合、[Step Description] も必要です。

## 受け入れテスト セッションの実行

Project Center には、手動テスト シナリオを保持するリポジトリがあります。手動テスト シナリオは、147 ページの「手動テスト シナリオの追加と編集」の説明にあるように定義されます。手動テスト シナリオは次の目的に有効です。

- シナリオに要求またはタスクを関連付け、その要求 / タスクに基づいて目的の機能を定義する。
- シナリオに欠陥を関連付け、その欠陥を修正する際、目的の機能を定義します。
- 受け入れテスト。受け入れテストの中で、一般的なシナリオはアプリケーションの機能を定義します。

プロジェクトの進行中に受け入れテストを作成して保守することは良いプラクティスです。そのような受け入れテストは、この特定のプロジェクト リリースだけでなく、プロジェクト ライフサイクル全体を通して開発されたプロジェクト機能を徹底的にテストすることを目的とします。

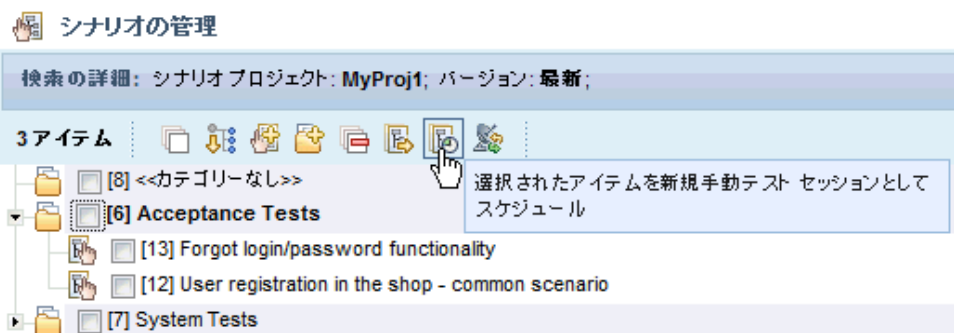
プロジェクトのリリースが近づいたら、管理者は Project Center リポジトリから適切な受け入れテストを選択し、テストの実施予定を立てることができます。

## 新しい手動テスト セッションのスケジューリング

プロジェクトの受け入れテスト（手動テスト セッション）を実行するには、次の操作を行います。

1. 次のいずれかの操作を行って、実行するテスト シナリオを指定します。
  - [テスト] ドロップダウン メニューの [管理] をクリックして [シナリオの管理] ページを表示します。シナリオ セットごとに並べられたプロジェクト シナリオのツリーを確認し、テスト セッションでスケジューリングするシナリオを選択します。そして [選択されたアイテムを新規手動テスト セッションとしてスケジュール] アイコンをクリックします。

図 16: [シナリオの管理] ページ



- [シナリオ] ページで、テスト セッションでスケジューリングするシナリオを選択し、[テスト セッションとしてスケジュール] ボタンをクリックします（このページは、特定の条件に一致するシナリオを検索した後に表示されます）。

- 表示された [ 手動テスト セッション作成 ] ダイアログで、テスト セッション名、開始日、および最終期限を指定します。

 全般

プロジェクト: MyProj1


名前:


説明:

ステータス: Draft

 スケジュール

反復 ID:  [\[検索\]](#) [\[追加\]](#)

開始予定日:  

最終期限:  

 スケジュール済みシナリオ

	担当者
Acceptance Tests	<input type="text"/>
Forgot login/password functionality	<input type="text"/>
User registration in the shop - common scenario	<input type="text"/>

- (オプション) 手動テストごとに、そのテストを実行するチーム メンバーを選択します。
- (オプション) 必要に応じて添付ファイルを追加します。
- [ 作成 ] ボタンをクリックします。

**注意:** テスト セッションが作成されるときに、割り当てられた担当者ごとに個別にタスクが作成されます。そのタスクには複数のテストが含まれることがあります。また、担当者が割り当てられていないすべてのシナリオに対して 1 個のタスクが作成されます。

テスト セッションが作成され、[ 手動テスト セッションの編集 ] ページが表示されます ( 図 17)。

図 17: [ 手動テスト セッションの編集 ] ページ

手動テスト セッションの編集: [4] manual\_1

全表示 欠陥 0 変更履歴

全般

プロジェクト: MyProj1

名前: manual\_1

説明:

ステータス: Draft

スケジュール

反復 ID: [検索] [追加]

開始予定日: [カレンダー]

最終期限: [カレンダー]

クローズ済み:

スケジュール済みシナリオ

2 アイテム 割り当て 割り当て解除

ID	名前	バージョン	タスク ID	担当者	シナリオ実行結果	実行開始日	実行終了日	継続時間	実行者	実行
12	User registration in the shop - common scenario	0	99	JohnSmith	Not run					[実行]
13	Forgot login/password functionality	0	99	JohnSmith	Not run					[実行]

シナリオ セットごとにタスクが作成され、タスク ID が表示されます。これで、テストを開始してこのセッションで実行することができます。

## スケジュールリングされたテスト セッションの変更

スケジュールリングされたテスト セッションを変更するには、次の操作を行います。

- [ テスト セッションの編集 ] ページを開きます (たとえば、[ テスト ] ドロップダウン メニューの [ テスト セッション ] を選択し、実行するテスト セッションの名前をクリックします)。
- 必要に応じてテスト セッションを変更します。たとえば、テスト セッションに対するシナリオの割り当てと割り当て解除、添付ファイルの追加と削除などを行うことができます。

テスト セッションに対する変更はすべて [ 変更履歴 ] タブに記録されます。

タスクは、変更に基づいて生成または変更されます。たとえば、あるシナリオをすでにユーザー 1 に割り当てていて、さらに別のシナリオをこのユーザー 1 に割り当てた場合、このテスト セッションを実行するタスクは、割り当てられた両方のシナリオを含むように変更されます。

## 割り当てられたテスト タスクの確認

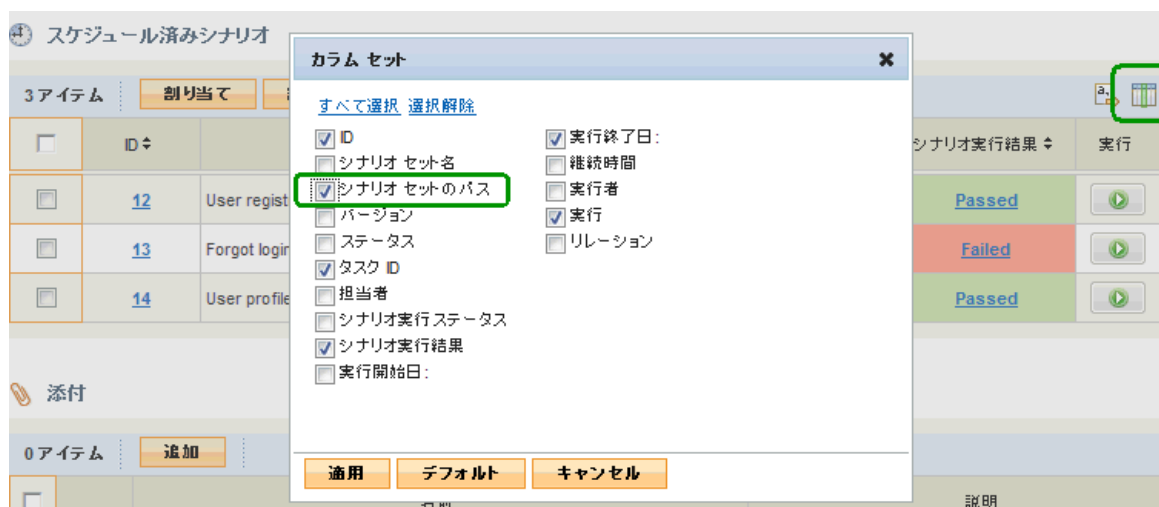
自分に割り当てられたテスト タスクを確認するには、次の操作を行います。

- Project Center で [ タスク ] タブをクリックし、選択したプロジェクトのタスク リストを開きます。

## シナリオへのパス

シナリオ テーブルを参照していて、シナリオ セットのツリー（つまり、Project Center のプロジェクトの [ テスト ] > [ 管理 ] ページに表示される完全なテスト ツリー中）のどこに特定のシナリオが「存在する」のかを知りたい場合、シナリオ テーブルの [ シナリオ セットのパス ] カラムでパス情報を知ることができます。このカラムはデフォルトでは表示されません。このカラムを表示するには次の操作を行います。

1. [ カラム セットの編集 ] ボタンをクリックします。
2. [ シナリオ セットのパス ] チェックボックスをオンにして [ 適用 ] をクリックします。



## 手動テスト セッションの開始

自分の手動テスト セッションのための手動テストは、テスト セッション レポート、またはタスク、要求、欠陥 / 改善のページから開始することができます。

### [ テスト セッション ] ページからの手動テストの開始

[ テスト セッション ] ページからテストを実行するには、次の操作を行います。



1. [ テスト セッションの編集 ] ページを開きます (たとえば、[ テスト ] ドロップダウン メニューの [ テスト セッション ] をクリックし、実行するテスト セッションの名前をクリックします)。

手動テスト セッションの編集: [3] Milestone 1 internal testing - user registration and profile functionality

全般 欠陥 0 変更履歴

We should perform final testing at the end of Milestone 1.

ステータス: Draft

スケジュール

反復 ID: 8 [検索] [追加]

開始予定日: 2012-10-03

最終期限: 2012-10-28

クローズ済み:

スケジュール済みシナリオ

3 アイテム 割り当て 割り当て解除

<input type="checkbox"/>	ID	名前	タスク ID	シナリオ実行結果
<input type="checkbox"/>	12	User registration in the shop - common scenario	97	Passed
<input type="checkbox"/>	13	Forgot login/password functionality	97	Failed
<input type="checkbox"/>	14	User profile settings page	97	Passed

2. [ スケジュール済みシナリオ ] セクションの [ 実行 ] ボタンをクリックします。

図 18: [ テスト セッションの編集 ] ページからの実行

シナリオ実行結果	実行終了日	実行者	実行
Passed	2012-10-05 18:25	admin	
Failed	2012-10-05 18:25	admin	
Passed	2012-10-05 18:25	admin	

3. 163 ページの「手動テストの実行」の説明にあるように、適切なテストを実行します。

## タスク、要求、欠陥 / 改善のページからの手動テストの実行

テスト セッションのすべてのテストを見ることができるので [ テスト セッションの編集 ] ページからテストを実行するのが最も便利ですが、[ タスクの編集 ] ページからテストを実行することもできます。次の操作を行います。

1. タスク、要求、または欠陥 / 改善の編集画面を開き、[ シナリオ ] タブをクリックします。
2. [ 実行 ] ボタンをクリックします。

図 19: [ シナリオ ] タブの例

タスクの編集: [32] Functional Test Execution

全般 変更履歴 コード 0 自動テスト 0 シナリオ 2/2

2 アイテム 新規 割り当て 割り当て解除

<input type="checkbox"/>	ID	名前	バージョン	シナリオ実行結果	テスト履歴	実行
<input type="checkbox"/>	16	func_2	0	(再)テストが必要		
<input type="checkbox"/>	15	function_1	0	(再)テストが必要		

3. 163 ページの「手動テストの実行」の説明にあるように、適切なテストを実行します。

#### 特定のテスト セッションが実行できない場合

テスト セッションのステータスが "Not Runnable" である場合、テストを実行することはできません。テストを実行しようとして次のメッセージが表示された場合、テスト セッションのステータスは "Not Runnable" です。

- 以下のテストは、タスクのステータスが「作業中」ではないため、実行できません。
- 以下のテストは、テスト セッションが実行可能な状態ではないため、実行できません。
- 以下のテストは、タスクのステータスが「作業中」ではなく、また、テスト セッションが実行可能な状態ではないため、実行できません。

## 手動テストの実行

いったん手動テストを開始したら、以下のようにテストを実行できます。

1. [手動テストの実行] ウィンドウの [実行] タブで [テストの開始] ボタンをクリックします。

図 20: [実行] タブ

実行 シナリオの添付ファイル 0

全般

名前: \* func\_2 - テスト 1

ステータス: Ready for Review

シナリオ: [16] func\_2

シナリオのバージョン: 0

シナリオの説明:

テスト セッション ID:

コメント:

結果: 未実行

実行: func\_2 - テスト 1

テストの開始

2. 最初のステップとして以下の操作を実行します。
  - a. 指定されたテスト アクションを完了します。
  - b. 実際の結果を指定します。
  - c. このステップに添付ファイルを追加したい場合（たとえば実際の結果の画面ショットなどを添付として追加したい場合）、[添付] フィールドのクリップのアイコンをクリックして添付ファイルを指定します。
  - d. (オプション) このテスト ステップに関連するコメントを入力します。
  - e. テストのステータスを更新します。

- f. [ 続行 ] をクリックします。

実行	シナリオの添付ファイル  0	欠陥  0	変更履歴
----	----------------	-------	------

**ステップ #1: F#を入力する**

**期待される結果:**

添付: (0 アイテム)

**実際の結果:** 「実践F#」と「F#入門」の2つがリストにあがる

**結果の添付ファイル:** (0 アイテム)

**コメント:** 検索スピード問題なし

**結果::** 成功

3. テスト ステップごとに上記の操作を繰り返します。
4. すべてのテスト ステップが完了したら、テスト全体についてのコメントを入力して [ テストの終了 ] をクリックします。

実行	シナリオの添付ファイル  0	欠陥  0	変更履歴
----	----------------	-------	------

**名前:** 検索テスト - テスト 1

**ステータス:** Ready for Review

**シナリオ:** [18] 検索テスト

**シナリオのバージョン:** 0

**シナリオの説明:** 本の検索を行う

**テスト セッション ID:**

**開始時刻::** 10/19/12 2:35 PM

**コメント:** 正常に検索および注文が終了した。 |

**結果::** 成功

## テスト結果の参照

テスト結果を参照するには、[ テスト ] ドロップダウン メニューの [ テスト セッション ] をクリックして [ テスト セッション ] レポートを表示し、適切なテスト セッションをクリックします。スケジュール済みシナリオ ] セクションに次の情報が表示されます。

ID	テスト セッションの ID。
名前	テスト セッションの名前。
タスク ID	テスト セッションのタスクの ID。
担当者	テスト シナリオの実行を割り当てられたユーザー。
ステータス	シナリオで指定された、全体的なシナリオのステータス (たとえばドラフト、実行準備完了、完了)。
シナリオ実行ステータス	特定のテスト実行のステータス。テストの実行中に指定されます。
シナリオ実行結果	シナリオが実行されたかどうかを示す自動生成の表示。
実行開始日	テストの実行が開始した時刻。
実行終了日	テストの実行が終了した時刻。
継続時間	開始時刻から終了時刻までの経過時間。
実行者	テストを実行したユーザー。
リレーション	関連する成果物 (タスク、欠陥など)。

与えられた情報に基づいて、QA チームのリーダーは最終期限の前にテストが完了するよう (たとえばタスクに割り当てる QA チーム メンバーの数を増やすなど)、チームの作業を計画して編集することができます。

テスト セッション レポートの詳細については 202 ページの「[ テスト セッション ] レポート」を参照してください。

Report Center の次の場所からもテスト結果を参照できます。

- Report Center > テスト > 手動テスト セッション: 202 ページの「[ テスト セッション ] レポート」を参照。
- Report Center > アーキテクト ダッシュボード > [ 手動テスト セッション ] レポート: 202 ページの「[ テスト セッション ] レポート」を参照。
- Report Center > テスター ダッシュボード > [ 手動テスト セッション ] レポート: 202 ページの「[ テスト セッション ] レポート」を参照。

## 再テストが必要かどうかの判断

Report Center の変更ベースのテスト レポートは、ソース コードの変更に応えて手動テストを再実行する必要があることをユーザーに通知します。

詳細については 105 ページの「変更ベースのテスト レポート」を参照してください。

## テスト セッションへの添付ファイルの追加

テスト ステップごとに添付ファイルを追加するだけでなく、テスト セッションのレベルで添付ファイルを追加することができます。次の操作を行います。

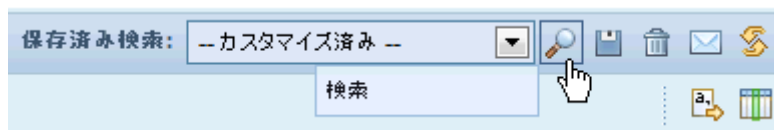
1. 添付ファイルを追加するテスト セッションの [ 手動テスト セッション ] ページを開きます。  
[ 添付 ] セクションで [ ファイルの選択 ] をクリックし、添付ファイルを追加します。

### MySQL データベースのデフォルトの最大パケット サイズは 1MB です

このデフォルト設定のために、添付ファイルの最大ファイル サイズは 1MB に制限されます。MySQL データベースの最大パケット サイズを変更せずに 1 MB より大きいファイルを添付しようとすると、エラーが発生する可能性があります。

## テスト セッションの検索

GUI 右上にある標準的な検索機能を使ってテスト セッションを検索できます。検索のヒントについては 51 ページの「Project Center での検索操作」を参照してください。



検索条件がプロジェクトの制限しかない場合、または制限がまったくない場合、検索結果のページに次のリンクが表示されます。

ID	名前
<a href="#">2</a>	システムテスト
<a href="#">1</a>	機能テスト
<a href="#">手動テスト セッションに所属していないシナリオ実行</a>	
合計	

どのテスト セッションにも属さないシナリオを参照するには、このリンクをクリックします。

## Project Center からの SOAtest の実行

Project Center インターフェイスから SOAtest のテストを実行できます。SOAtest のテストは、[ シナリオの管理 ] ページにインポートし、手動テスト セッションにスケジューリングし、他の Project Center のテスト シナリオと同じように実行できます。

現在、**Parasoft Development Testing Platform はアクセス制御を必要とする SOAtest Server へのアクセスをサポートしていません。** Development Testing Platform が SOAtest Server にアクセスする場合、アクセス制御を無効にした SOAtest Server を使用する必要があります。

### 前提条件

この機能は SOAtest 9.1 以上でサポートされます。

テストを実行する前に、必ず次のことを確認してください。

- SOAtest Server の Web サービスが実行中であり、テストがあるワークスペースに対して実行するように設定されている。
- 284 ページの「SOAtest サーバーの設定」の説明にあるように、SOAtest Server の設定が管理者によってすでに設定されている。

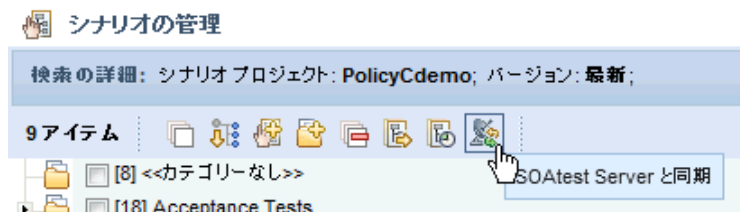
### Project Center のテストと SOAtest ワークスペースの同期化

Project Center では、ワークスペースに現在ある SOAtest テストに自動的にアクセスしてテスト シナリオを設定することができます。

次の操作を行います。

1. [ テスト ] メニューの [ 管理 ] をクリックします。

[ シナリオの管理 ] ページで、[ SOAtest Server と同期 ] ボタンをクリックします。



2. [ SOAtest Server の照会 ] をクリックします。Project Center が SOAtest のワークスペースに接続します。以下の 3 つのカテゴリに属する SOAtest .tst ファイルのリストが表示されます。

- **作成されるテストスイート:** [ シナリオの管理 ] ページにまだ存在しない SOAtest ワークスペースの .tst ファイル。
  - **更新されるテストスイート:** [ シナリオの管理 ] ページにすでに存在する SOAtest ワークスペースの .tst ファイル。
  - **同期されたテストスイート:** [ シナリオの管理 ] ページにすでに存在していて、SOAtest ワークスペースから更新する必要がない .tst ファイル。
3. [ 同期 ] をクリックします。[ シナリオの管理 ] ページに、ワークスペースにあるすべての SOAtest .tst ファイルに対する現行テスト シナリオが表示されます。個々のテスト シナリオは、そのテスト シナリオ名の元になった .tst ファイルを実行するように自動的に構成されます (つまり、各テスト シナリオは 1 つのテスト ステップを持ち、このテストステップは対応する SOAtest .tst ファイルを実行するように設定されます)。SOAtest Server のプロパティ、テストスイートのパラメーター、およびソース管理のパラメーターがテスト シナリオに追加されます (ソース管理のパラメーターは SOAtest がソース管理システムに接続している場合に利用できます)。

シナリオの編集: [17] Test1.tst

シナリオ 要求 0 欠陥 0 添付 0 テスト履歴 0 変更履歴

**全般**

プロジェクト:

名前:

説明:

ステータス:

シナリオセット ID:

**シナリオのステップ**

ステップ #1: SOAtest の呼び出し

**SOAtest Server のパラメーター**  
 ホスト: 172.24.254.14  
 ポート: 9080

**SOAtest Server テストスイートのパラメーター**  
 パス: /Tutorial1/Test1.tst  
 コンフィギュレーション: soatest.user://MyWebFuncTest  
 環境:

**ソース管理パラメーター**  
 リビジョン:



ここまでの操作が終了すると、他のテスト シナリオと同じようにテストを編集してスケジューリングできるようになります。

## SOAtest を起動するテストの実行

SOAtest のテスト ステップを含むテストを実行するには、次の操作を行います。

1. 他の手動テスト シナリオの場合と同じようにテストを開始します ( 詳細については 158 ページの「受け入れテスト セッションの実行」を参照してください)。
2. SOAtest を起動するテスト ステップに到達したら、必要に応じて SOAtest のパラメーターを変更し、[SOAtest の呼び出し] ボタンをクリックします。SOAtest が起動されます。
3. 実行が完了したら、[SOAtest の結果を取得] ボタンをクリックします。

期待される結果:

添付: (0 アイテム)

SOAtest の実行: 実行 ID: 482861928 SOAtest の結果を取得 待機する秒数: 30  
 実行の進捗: 0 %

実際の結果:

結果は [ 実行の結果 ] に記録されます。[ 結果の添付ファイル ] のコントロールから結果の完全なレポートにアクセスすることができます ( ペーパークリップのアイコンをクリックすると表示されます )。

期待される結果:

添付: (0 アイテム)

SOAtest の実行: SOAtest の呼び出し

実際の結果: SOAtest の実行結果, Thu Oct 18 11:52:51 JST 2012  
 5 のうち 5 テストが成功しました。0 タスクがレポートされました。

結果の添付ファイル: 📎 (1 アイテム)

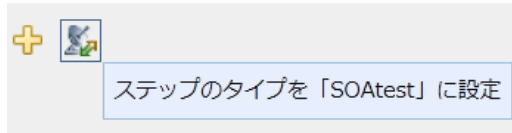
1 アイテム 追加 削除

<input type="checkbox"/>	名前	説明	添付日
<input type="checkbox"/>	 <a href="#">step1.html</a>	SOAtest の実行結果	2012-10-18 11:53

## テストステップとしての SOAtest テストの追加

複数のステップがあるテスト シナリオを手動で定義していて、そのうちの 1 つのステップとして SOAtest のテストを実行したい場合もあります。たとえば、テスト シナリオが手動テストと自動テストの組み合わせである場合などです。その場合、SOAtest を起動するステップで以下の操作を行います。

1. [ ステップのタイプを「SOAtest」に設定 ] ボタンをクリックします。



Project Center 管理者によって設定された SOAtest Server の設定がコントロールに取り込まれます。

2. この設定を上書きしたい場合 (たとえば別のテスト コンフィギュレーションを実行する必要がある場合)、必要に応じて設定を変更します。
3. [ パス ] フィールドで、実行する .tst ファイルへのパスをワークスペースへの相対パスで指定します。

SOAtest Server テストスイートのパラメーター	
パス:	/Special/SpecialTest.tst
コンフィギュレーション:	soatest.user://MyWebFuncTest
環境:	

# Project Center のレポート

このセクションの内容：

- プロジェクト ステータス レポート
- 要求レポート
- 欠陥 / 改善レポート
- タスク レポート
- イテレーションレポート
- テスト レポート

# プロジェクトステータス レポート

Project Center のレポートは、さまざまな角度からプロジェクト成果物（要求、欠陥、タスク）を視覚化します。プロジェクトステータス レポートは、プロジェクトライフサイクルのどの段階でも幅広い情報を提供します。以下はこのレポートをいつどのように使用するかの例です。

- **計画のため：**すべての計画済みイテレーションとタスクに加えて、推定終了日、タスクとしてスケジュールされたすべての欠陥、実装が予定されているすべての要求が表示されます。
- **進行中のプロジェクトのため：**タスク、欠陥、要求の作業進捗をモニターできます。
- **プロジェクトの解析とプロセス改善のため：**プロジェクト成果物に関するさまざまな履歴データが表示されるので、将来のプロジェクトでプロセスの改善が可能かどうかを判断できます。

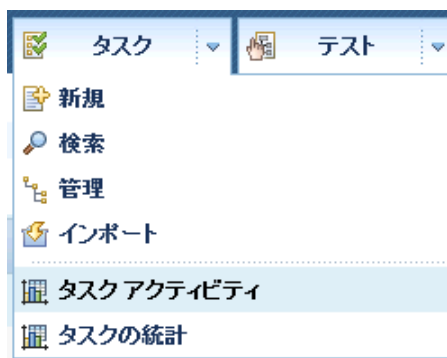
以下の種類のプロジェクトステータス レポートを利用できます。

- [ 要求のステータス ] レポート
- [ 欠陥 / 改善のステータス ] レポート
- [ イテレーションのステータス ] レポート
- [ タスクの統計 ] レポート
- [ タスク アクティビティ ] レポート

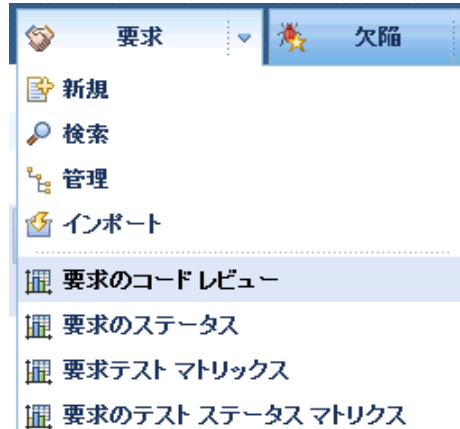
これらのレポートには以下の 2 つの方法でアクセスできます。

- [ レポート ] メニューから
- 特定の成果物のメニューから

たとえば、[ タスク ] メニューからは 2 種類のタスク関連のレポートを開くことができます。



同様に、[ 要求 ] メニューからは 4 種類の要求関連のレポートを開くことができます。



---

## [ 要求のステータス ] レポート

[ 要求のステータス ] レポートには、自分の要求の実装ステータス (完了、作業中、スケジュールなし)、および各要求のテストステータスが表示されます。詳細については 180 ページの「[ 要求のステータス ] レポート」を参照してください。

---

## [ 欠陥 / 改善のステータス ] レポート

[ 欠陥 / 改善のステータス ] レポートには、選択したプロジェクトの欠陥と改善が表示されます。欠陥 / 改善の現在のステータス、アイテムごとのテスト結果、および履歴トレンドグラフが表示されます。詳細については 187 ページの「[ 欠陥 / 改善のステータス ] レポート」を参照してください。

---

## [ イテレーションのステータス ] レポート

[ イテレーションのステータス ] レポートには、選択したプロジェクトのイテレーションとタスクについての情報が表示されます。詳細については 197 ページの「イテレーションレポート」を参照してください。

---

## [ タスクの統計 ] レポート

[ タスクの統計 ] レポートには、タスクの実装のステータスが表示されます。詳細については 191 ページの「[ タスクの統計 ] レポート」を参照してください。

---

## [タスク アクティビティ] レポート

[タスク アクティビティ] レポートには、特定の期間にプロジェクトで作業されたタスクのリストが表示されます。詳細については 192 ページの「[タスク アクティビティ] レポート」を参照してください。

---

# 要求レポート

以下の要求レポートを利用できます。

- [ 要求のコード レビュー ] レポート
- [ 要求のコード レビュー ] レポート
- [ 要求のステータス ] レポート
- [ 要求テスト マトリックス ] レポート

---

## [ 要求バーンダウン ] レポート

[ 要求バーンダウン ] レポートは、選択された要求がどの程度完了に近づいているかを表します。プロジェクトの経過に対する完了したストーリー ポイントの割合に関する情報を表示し、作業の完了状況の傾向を把握するのに役立ちます。

検索条件で要求を指定することによって、特定の要求だけをレポートに含めることができます。また、すべての要求のレポートを生成することもできます。

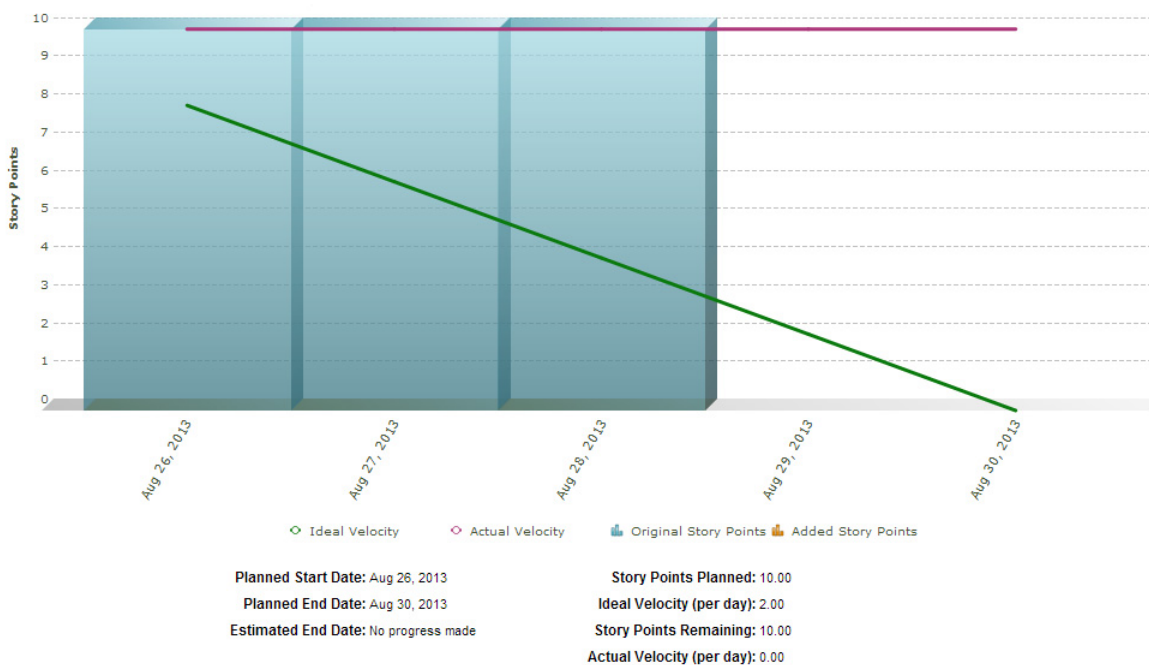
プロジェクト開始日から終了日までの期間が X 軸にプロットされます。プロジェクトの開始日および終了日は、Project Center の管理ページで変更できます。詳細については 230 ページの「プロジェクトの作成と構成」を参照してください。ストーリー ポイントは Y 軸にプロットされます。

---

## バーンダウンレポートのチャート

バーンダウンレポートのチャートは、プロジェクトの毎日の進捗を一目で把握するのに役立ちます。

- [ 元々のストーリーポイント ] バー ( 青色 ) は、プロジェクト開始時点で要件に割り当てられていたストーリー ポイントを表します。
- [ 追加のストーリーポイント ] バー ( 茶色 ) は、プロジェクト開始日以降に追加されたポイントを表します。
- [ 理想のベロシティ ] 折れ線 ( 緑色 ) は、プロジェクト開始日から終了日に均等にストーリー ポイントが配分された場合、完了しているべきポイントの割合を表します。理想のベロシティに近い割合で作業が完了している場合、チームはプロジェクトをスケジュール通りに完了するために理想的な位置にいます。
- [ 実際のベロシティ ] 折れ線 ( 紫色 ) は、現在のデータに基づいて、実際に完了済みの日ごとのストーリー ポイントを表します。



ストーリー ポイントがバーンダウン チャートに表示されるには、要求が未解決のステータスでなければなりません。デフォルトでは、Development Testing Platform の要求には「レビュー」と「完了済み」の 2 つのステータスがあります。どちらのステータスも「未解決 - 編集可能」として設定されています。デフォルトのステータスを変更したり、他のステータスを追加するには、次の操作を行います。

1. Development Testing Platform メイン メニュー バーの [ 管理 ] をクリックします。
2. [ 設定 ] > [ 成果物のステータス ] をクリックします。
3. [ 要求のステータス ] セクションで [ 新規 ] をクリックします。
4. ステータスの名前を入力し、[ 状態 ] ドロップ ダウン メニューから状態を選択して [ 保存 ] をクリックします。

成果物のステータスに関するその他の情報については、274 ページの「カスタム ステータスの追加」を参照してください。

Requirement Statuses		
3 Item(s) <span style="float: right;">New</span>		
Active	Name	State
<input checked="" type="checkbox"/>	Review	Unresolved - Editable
<input checked="" type="checkbox"/>	Finalized	Unresolved - Editable
<input checked="" type="checkbox"/>	Implemented	Resolved - Editable



[要求バーン ダウン] レポートは、タスク指向ではありません。プロジェクトに関連付けられたタスクのステータスは、要求のステータスに影響を与えません。最終的なストーリー ポイントの値を計算する際は、要求のステータス（解決済み / 未解決）だけが考慮されます。

---

## バーン ダウン レポートのその他の情報

チャートの情報に加えて、[要求バーン ダウン] レポートには、プロジェクト、要求、およびストーリー ポイントに関わる重要な統計情報が表示されます。

- 開始予定日および終了予定日：プロジェクトの管理ページで定義されたプロジェクトの開始日および終了日です。詳細については 230 ページの「プロジェクトの作成と構成」を参照してください。
- 見込みの終了日：日ごとのバーンダウンに基づいて推測された、選択された要求がすべて完了する予定日です。
- 計画されたストーリーポイント：選択された要求のストーリー ポイントの合計です。
- 理想のベロシティ (1 日当り)：プロジェクトをスケジュール通り完了するために日ごとに完了すべき理想的なストーリー ポイントです。
- 残ストーリーポイント：未解決の要求のストーリー ポイントです。
- 実際のベロシティ (1 日当り)：実際に完了された日ごとのストーリー ポイントの平均です。

---

## [要求のコード レビュー] レポート

ピア コード レビューのプロセスは、Parasoft Test 製品（たとえば Jtest、C++test、dotTEST、SOAtest）で使用できるコード レビュー機能を介して自動化することができます。Development Testing Platform は、これらのツールのコード レビュー データを格納し、さまざまなコード レビュー レポートを提供します。このレポートによって、ユーザーはコード レビューの傾向を時間とともにビジュアル化することができます。

[要求のコード レビュー] レポートは、要求とタスクの観点からコード レビュー プロセスの可視性を提供します。このことは「要求が実装される時、関連するコードをレビューしなければならない」といったポリシーがある場合に特に役立ちます。

特定の要求だけに集中したい場合などには、要求の検索条件によってレポートの内容をフィルタリングすることができます。このレポートは、要求の階層リストおよび関連するタスク（親子関係）を表します。成果物（要求またはタスク）ごとに、現在のコード レビューのステータスを参照できます。これには、以下の状態のコード レビュー スレッドの数も含まれます。

- レビューアーがレビューを実行するのを待っている保留中のコード レビュー
- レビューアーが完了したコード レビュー
- オープン中の発見されたコード レビューの問題
- 発見され、修正されたコード レビューの問題

ID	Name	Code Reviews		Issues	
		To Review	Done	To Fix	Done
3384	(V49R5) Requirement Specification report	0	0	0	0
43476	Requirments Specification - export to MS Word	0	0	0	0
43477	Requirments Specification - report	0	0	0	0
3557	(V49R1) New Code Review report	117	72	0	0
3669	(V49R3) France - minor improvements to Report Center	0	15	0	0
3679	(V49R4) Tasks dependencies and hierarchy	112	0	0	0
3708	Celgene High Priority - Concerto 4.9	21	55	0	3
3728	Performance improvement for Concerto 4.9	10	126	0	0
Total		260	238	0	3

このような結果から、ユーザーは以下のことを判断できます。

- まだコードがレビューされていない要求あるいはタスクがあるかどうか。
- 特定の要求について、レポートされた「コード レビューの課題」の数が平均よりも多くないかどうか。
- 特定の機能を検証するために何個のコード レビューが実行されたか。

タスクまたは要求の名前をクリックすると、[コード レビューの詳細] レポートが開き、コード レビュー プロセスの詳細が表示されます。

#### 要求のコードレビューの詳細

検索の詳細: ID: 91;

ソースの表示 スレッドの表示

- 91 User registration and profile 0 | 0
  - 93 Easy user registration in the shop 0 | 0
    - 108 Easy user registration in the shop - server side 0 | 0
    - 109 Easy user registration in the shop - GUI front-end 0 | 0
    - 143 Requirement Review: Easy user registration in the shop 0 | 0
    - 145 Requirement Review: Easy user registration in the shop 0 | 0
  - 92 Extra security needed 0 | 0
    - 131 Implement 要件 92 : Extra security needed - HTTPS Support 0 | 0
    - 132 Extra security needed - Automated logout after 5 minutes idle time 0 | 0

詳細については [コードレビューの詳細] レポートを参照してください。

## コード レビュー プロセスの設定と [要求のコード レビュー] レポート

[要求のコード レビュー] レポートにデータが確実に表示されるようにするために、ユーザーが特別な Parasoft Test の設定を入力する必要はありません。ただし、以下のワークフローは理解して実践する必要があります。

Parasoft Test 製品がコード レビュー解析を実行してコード レビュー タスクを作成する場合：

- **コミット後コード レビューの場合**：Parasoft Test は、このファイルとリビジョンが タスクに関連付けられているかどうかを Development Testing Platform に問い合わせます。

- **コミット前コード レビューの場合**：ダイアログが表示され、Parasoft Test ユーザーは特定のコード変更に関連付けられているタスク ID を入力できます。

コード レビューがこのようにタスクに相互に関連付けられている場合、Parasoft Test はこのコード レビュー タスクを、関連付けられているタスクにリンクします。タスクが要求に関連付けられている場合、そのタスクは [ 要求のコード レビュー ] レポートに 表示されます。

## 注意

Parasoft Test 製品からコード レビューを実行すると、要求ではなくタスクにコード レビューが関連付けられます。コード レビューがどのように要求と関連付けられているかを見るには、[ 要求のコード レビュー ] レポートを利用します。

[ 要求のコード レビュー ] レポートでは、コード レビューが実行されてコード レビュー タスクが作成される前に、タスクとコードの関連付けを追加する必要があります。これは次のことを意味します。

- チームが **ポスト後コード レビュー** を実行し、ファイルをコミットするときに @task を指定する場合、タスクとファイル リビジョンの関連付けを Development Testing Platform が認識するよう、コード レビューを実行する前に Source Scanner を実行する必要があります。
- チームが **ポスト前コード レビュー** を実行する場合、Source Scanner は必要ありません。なぜなら、タスクの関連付けは Parasoft Test インターフェイスを介してユーザーが指定するからです。

コード レビュー スキャンの後にコードとタスクの関連付けを行う場合、コード レビュー タスクは関連するタスクを参照せず、[ 要求のコード レビュー ] レポートにも表示されません。

---

## [ 要求のステータス ] レポート

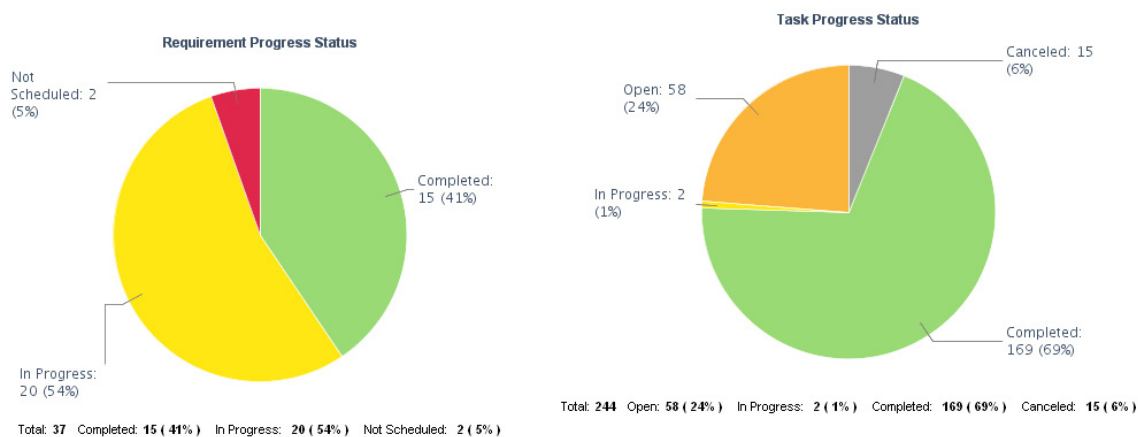
このレポートは、要求の実装ステータス（完了、開発中、スケジュールなし）に関する情報を表示します。実装ステータスは、要求に関連付けられているタスクに基づきます。このレポートのデータは csv または xml 形式でダウンロードすることができます。

---

### 要求のステータス

このレポートには 2 つのチャートがあり、プロジェクトの要求に関するメトリクスの概要を視覚的に表示します。

- **要求進捗ステータス**：完了済みの要求、作業中の要求、スケジュールされていない要求の割合を表示します。
- **タスク進捗ステータス**：プロジェクトのすべての要求に関連付けられたすべてのタスクのステータスを表示します。



ID	Name	Priority	Progress	Planned Time	Actual Time	Remaining Time	Tasks			
							Open	In Progress	Completed	Canceled
3653	9.2 Service Packs	50000	100%	85d 0h	110d 0h	0d 0h	0	0	67	1
3658	Ability for SOAtest and Virtualize to be Installed Together	50000	0%	24d 0h	0d 1h	23d 2h	14	0	0	0
3662	Change Management	50000	69%	178d 0h	117d 0h	49d 0h	27	1	63	14

表は、要求のステータスを完了率で表すとともに、要求を満たすために必要な見積り時間を表示します。

- 要求ステータスの詳細の進捗バーの数字は以下の時間計算によって計算されます。  
 $\% \text{ 進捗} = \text{総作業時間} / (\text{総作業時間} + \text{総残り時間})$
- 完了したタスクの場合**
  - 総作業時間 = 実際の時間
  - 総残り時間 = 0
- オープンだがまだ着手されていないタスク (つまり 実際の時間 = 0) の場合**
  - 総作業時間 = 0
  - 総残り時間 = 計画時間
- オープンまたは作業中のタスクで、すでに着手されたタスク (つまり 実際の時間 > 0) の場合**
  - 総作業時間 = 実際の時間
  - 総残り時間 = 残り時間

## 要求のテストステータス

この表は、各要求のテストステータスを表します。自動テストおよび手動テスト実行の両方の概要が表示されます。

ID	Name	Total Tests	Automated Tests			Manual Tests						Test Success Ratio
			Total	Passing	Failing	Total	Passing	Failing	In Progress	Incomplete	Not Run	
1245	Parasoft Enterprise Solution "usability" improvements	104	101	42	59	3	1	1	0	0	1	41%
1255	Concerto 10 Policy Center	1	1	1	0	0	0	0	0	0	0	100%
1259	Concerto 10 Manual Tests	140	117	58	59	23	21	1	1	0	0	56%
1270	Concerto organization improvement	0	0	0	0	0	0	0	0	0	0	Not Tested
1276	Concerto 10 requirement improvements	27	0	0	0	27	23	3	1	0	0	85%
1285	MassMutual: Filing test failures as defects into Quality Center	5	1	0	1	4	4	0	0	0	0	80%
1361	Polish Roadmap Requirements	80	67	60	7	13	9	3	0	0	1	86%
1623	Environment Manager & Provisioning	5	0	0	0	5	5	0	0	0	0	100%
1901	Celgene - Top level Requirement	193	44	36	8	149	122	22	1	1	3	82%
1903	Concerto 10 - General	14	0	0	0	14	12	2	0	0	0	86%
2033	Inomed top level requirement	6	0	0	0	6	4	1	0	0	1	67%
2061	TechMatrix top level requirement	2	0	0	0	2	2	0	0	0	0	100%

## [ 要求テストマトリックス ] レポート

[ 要求テストマトリックス ] レポートには、要求とそれに関連するテストが表示されます。要求からテストへのトレーサビリティが提供されます。

要求の検索条件を変更することで、任意の要求セットに対して [ 要求テストマトリックス ] レポートを生成することができます。デフォルトでは、プロジェクトのすべての要求が表示されます。

Requirements							Tests
ID ^	Name	Customer	Merge Status	Risk ID	Tag	Target Version	Name
787	[COV-GEN-01] Coverage instrumentation					--Not specified--	
788	[COV-GEN-02] Levels of the coverage reports					--Not specified--	
789	[COV-GEN-03] Coverage metrics configuration					--Not specified--	
790	[COV-GEN-04] Context of the coverage analysis					--Not specified--	
791	[COV-GEN-05] Coverage highlights in the source code					--Not specified--	
792	[COV-GEN-06] Traceability between coverage and tests					--Not specified--	
793	[COV-GEN-07] Coverage report					--Not specified--	
794	[COV-STMT-01] Statement coverage for expression statements					--Not specified--	
795	[COV-STMT-02] Statement coverage for iteration statements					--Not specified--	
796	[COV-STMT-03] Statement coverage for jump statements					--Not specified--	
797	[COV-STMT-04] Statement coverage for declaration statements					--Not specified--	
798	[COV-STMT-05] Statement coverage for conditional statements					--Not specified--	
799	[COV-STMT-06] Statement coverage for block statements					--Not specified--	
800	[COV-DEC-01] Decision coverage for conditional statements					--Not specified--	
801	[COV-DEC-02] Decision coverage for iterational statements					--Not specified--	
802	[COV-DEC-03] Decision coverage for switch statements					--Not specified--	
803	[COV-MCDC-01] MC/DC decisions for C++ boolean expressions					--Not specified--	
804	[COV-MCDC-02] MC/DC decisions for C conditional statements					--Not specified--	

[ 要求テストマトリックス ] レポートは次の情報を提供します。

- 要求のリスト：要求の ID、名前、および定義されたカスタム フィールド。

- テスト：関連するテストの名前。

このマトリックスは任意のカラムでソートできます。たとえば [ テスト名 ] でソートすると、まだテストが割り当てられていない要求を探すのに役立ちます。

[ 要求テスト マトリックス ] レポートは、MS Word の docx フォーマットにエクスポートしたり、印刷したり、保存することができます。この機能は特に認証監査 / コンプライアンス監査のために準備するときに役立ちます。

**注意**

エクスポートされた .docx ファイル フォーマットは、Microsoft Office 2007 以降または Microsoft Word Viewer で開くことができます。 .docx ファイル フォーマットは、Open Office または LibreOffice ではサポートされていません。

## [ 要求のテスト ステータス マトリックス ] レポート

[ 要求のテスト ステータス マトリックス ] レポートは、プロジェクトの要求と、要求に関連付けられたテストおよびテスト実行ステータスを表示します。このレポートは、要求からテストおよびテスト実行ステータスへのトレーサビリティを提供します。検索条件で指定された任意の要求のレポートを生成できます。以下の情報が表示されます。

- 要求の一覧：ID、オリジナル ID、説明
- テスト：テスト ステータス ( 下の「 [ テスト ステータス ] 列」を参照 )、シナリオ、シナリオ実行結果、テスト セッション

Tests			
Test(s) Status	Scenario	Scenario Run Result	Test Session
Failed	[21503] Test Saved Search ver. 8	Failed	
	[24515] Test Entity Create button ver. 2	Passed	[662] Concerto 4.8.4 release
	[24197] Requirement Test Status Matrix ver. 12	Passed	
	[24650] Performance of Scenario Run ver. 9	Failed	[678] Concerto 4.9 - US team
Not Tested	[25048] aaa ver. 0	Not Run	
	[21493] Defect association with Scenario, Scenario Run, Test Session ver. 3	Passed	[678] Concerto 4.9 - US team

以下のアクションを実行できます。

- 列のヘッダーをクリックしてマトリックスをソートします。
- レポートを印刷したり、Excel のスプレッドシート、Word 文書、または CSV ファイルとしてエクスポートできます。この機能は特に認証監査 / コンプライアンス監査のために準備するときに役立ちます。
- 要求 ID をクリックし、要求を参照および編集します。

- テスト ID をクリックし、テストを参照および編集します。
- テスト セッション ID をクリックし、テスト セッションを参照します。

## [ テスト ステータス ] 列

この列は、要求に割り当てられた手動テストシナリオの観点から、全体的なテスト ステータスを表します。この列は、要求に関連付けられたすべてのシナリオおよびその実行結果から導き出された完了ステータスを表示します。要求には複数のシナリオを割り当てることができるため、[ シナリオ実行結果 ] とは異なるステータス（成功、失敗、作業中または未実行）が表示される場合があります。次の表は、各ステータスの状態を表しています。

ステータス	ステータスが表示される場合...
未テスト	要求に関連付けられたシナリオのいずれかが「未実行ステータス」の場合、他のシナリオが失敗、未完了、作業中、成功である場合も、このステータスが表示されます。
失敗	要求に関連付けられたシナリオのいずれかが「失敗」ステータスの場合、他のシナリオが未完了、作業中、成功である場合も、このステータスが表示されます。「未テスト」ステータスのシナリオがある場合、「未テスト」ステータスが表示されます。
未完了	要求に関連付けられたシナリオのいずれかが「未完了」ステータスの場合、他のシナリオが未完了、作業中、成功である場合も、このステータスが表示されます。ただし、「未テスト」または「失敗」ステータスのシナリオがある場合、「未完了」ステータスは表示されません。
作業中	要求に関連付けられたシナリオのいずれかが「作業中」ステータスの場合、他のシナリオが成功である場合も、このステータスが表示されます。「未テスト」、「失敗」、または「未完了」ステータスのシナリオがある場合、「成功」ステータスは表示されません。
成功	要求に関連付けられたすべてのシナリオが「成功」ステータスである場合に、このステータスが表示されます。

## [ 要求仕様 ] レポート

このレポートは、指定された検索条件に一致する要求の詳細を表示します。レポートを印刷したり、Microsoft Word フォーマットでエクスポートすることができます。カスタム検索条件を使用してレポートを生成できます。

Requirement Specification

Search Details: Projects: Empty Project, Saved Search: -- CUSTOMIZED --

Sort by: Name ^ Original ID

**17 (3601) Browsing through the shop products**

Status: Review	Rank: 50000
Owner:	Created on: 10/21, 2012
Due Date:	Path:

**Description:**  
It should be easy and intuitive for customers to browse, search and get info about the products in our shop. This should be achieved in several ways: - easy quick search - detailed search - site map - browsable catalog - my favourite categories tagging - new items notification emails. See the sub-requirements for details.

**4 (3599) Easy user registration in the shop**

Status: Review	Rank: 50000
Owner:	Created on: 10/21, 2012
Due Date:	Path: 2

**Description:**  
It should be comfortable and almost seamless for the user to register in our shop. Let's require only the most basic data like email. The geographical location should be drawn from internet traffic data.

**3 (3598) Extra security needed**

使用例：プロジェクトの特定のリリース / イテレーションに割り当てたすべての要求の詳細なレポートを生成します。

要求ごとに以下の情報がレポートされます。

- ID
- 名前
- ステータス
- 所有者
- 期限
- ランク
- 作成日
- パス
- 説明
- 添付ファイル名



---

# 欠陥 / 改善レポート

以下の 欠陥 / 改善 レポートを利用できます。

- [ 欠陥 / 改善の統計 ] レポート
- [ 欠陥 / 改善のステータス ] レポート

---

## [ 欠陥 / 改善の統計 ] レポート

[ 欠陥 / 改善の統計 ] レポートは、さまざまな角度から欠陥 / 改善をビジュアル化します。以下の項目ごとに欠陥 / 改善をビジュアル化することができます。

- コンポーネント
- クローズ バージョン
- 存続期間
- オーナー

欠陥 / 改善の統計に関連するレポートにアクセスするには、[ 欠陥 ] タブのドロップダウン メニューから [ 欠陥 / 改善の統計 ] を選択します。

以下のレポートを参照できます。

- コンポーネントごとの欠陥
- クローズ バージョンごとの欠陥
- 存続期間ごとの欠陥
- オーナーごとの欠陥

---

## コンポーネントごとの欠陥

このグラフは、開発チームが分類のために使用しているコンポーネント（たとえばモジュール 1、モジュール 2 など）ごとに欠陥 / 改善をビジュアル化します。アクション ステータス（たとえば、修正でリリース待ち、修正で再テスト待ちなど）に対して、コンポーネントごとに欠陥 / 改善の数をレポートします。

---

## クローズ バージョンごとの欠陥

このグラフは、開発チームが使用している "クローズ バージョン" オプション（たとえば 1.0、1.1、2.0、2.1 など）ごとに欠陥 / 改善をビジュアル化します。アクション ステータス（たとえば、修正でリリース待ち、修正で再テスト待ちなど）に対して、クローズされた欠陥 / 改善の数をバージョンごとにレポートします。

---

## 存続期間ごとの欠陥

このグラフは、欠陥 / 改善の存続期間（システムに入力されてからの時間）ごとに欠陥 / 改善をビジュアル化します。アクション ステータス（たとえば、修正でリリース待ち、修正で再テスト待ちなど）に対して、存続期間ごとに欠陥 / 改善の数をレポートします。

---

## オーナーごとの欠陥

このグラフは、欠陥の修正に割り当てられたユーザーごとに欠陥 / 改善をビジュアル化します。欠陥 / 改善の現在のステータスに対して、欠陥 / 改善の数をレポートします。

---

## [ 欠陥 / 改善のステータス ] レポート

欠陥 / 改善レポートには、選択したプロジェクトについてレポートされた欠陥と改善が表示されます。欠陥 / 改善の現在のステータス、アイテムごとのテスト結果、および履歴トレンド グラフが表示されます。このレポートは次の 2 つのセクションから構成されます。

- 概要
- 欠陥の詳細と改善の詳細

---

### 概要

概要セクションには、選択プロジェクトについて欠陥と改善のステータスの数が示されます。表、欠陥グラフ、改善グラフに欠陥 / 改善についての概要が表示されます。

概要の表 ( 図 21 ) には、「Report Center ステータス」と呼ばれる以下のステータスが、選択したプロジェクトの欠陥と改善の両方について表示されます。

図 21: 欠陥 / 改善の概要

	合計	未解決	未テスト	失敗したテスト	修正済み
欠陥	42	37	0	0	5
改善	8	8	0	0	0

Report Center ステータスを含めて、以下のデータが欠陥 / 改善の概要として表示されます。

- **合計**：最終ドロップで使用できるステータスの数。
- **未解決**：バグ トラッキング システム中にある、まだ修正されていない欠陥 / 改善。
- **未テスト**：バグ トラッキング システム中にある、修正された欠陥 / 改善で、Report Center/ Project Center がテストを受け取っていない欠陥 / 改善。
- **失敗したテスト**：バグ トラッキング システム中にある、修正された欠陥 / 改善。さらにテストが実行されて Report Center/ Project Center はテスト結果を受け取ったが、テストは失敗だった。
- **修正済み**：バグ トラッキング システム中にある、修正された ( 解決された ) 改善 / 欠陥。さらに、テストが実行され、テスト結果を Report Center/ Project Center は受け取った。テストは成功だった。

欠陥グラフ ( 図 22 ) と改善グラフ ( 図 23 ) には、最後の 30 ドロップ ( ステータスが変わった日 ) での欠陥 / 改善ステータスの傾向が表示されます。

図 22: 欠陥グラフ

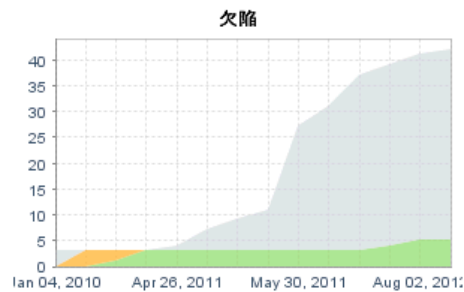


図 23: 改善グラフ



**注意：** 概要、プロジェクト、および他のシステム プロパティを含めて、検索条件に基づいて欠陥または改善を探す方法については 93 ページの「欠陥 / 改善」を参照してください。

## 欠陥の詳細と改善の詳細

欠陥の詳細と改善の詳細の表は、個々の欠陥 / 改善についての情報を提供するという点で互いに似ています。この情報には、テスト ステータスおよびタスクとの関係も含まれます。図 24 は欠陥の詳細の表です。

図 24: 欠陥の詳細

欠陥										
	ID	作成日	概要	優先度	重要度	割り当て:	ステータス	Report Center ステータス	テストが必要	
1	<a href="#">54</a>	2012-08-09	characters	Medium	Medium		New	未解決	はい	
2	<a href="#">50</a>	2012-08-02	Forgot password reactivation email functionality does not work	High	High	JohnSmith	Assigned	未解決	はい	
3	<a href="#">51</a>	2012-08-02	National characters do not work in user profile data	Medium	Medium	DavidScott	Resolved	修正	はい	<a href="#">111</a>
4	<a href="#">46</a>	2012-06-12	Forgot password reactivation email functionality does not work	High	High	JohnSmith	Assigned	未解決	はい	
5	<a href="#">47</a>	2012-06-12	National characters do not work in user profile data	Medium	Medium	DavidScott	Resolved	修正	はい	<a href="#">61</a>

欠陥 / 改善の詳細の表は、欠陥 / 改善ごとに以下の情報を提供します。

- **ID:** 欠陥 / 改善に割り当てられた、固有の識別番号。ID をクリックすると、[View Defect/Enhancement] ページが開きます。選択した欠陥 / 改善についての詳細情報が表示されます。
- **作成日:** 欠陥 / 改善が作成された日。
- **概要:** 欠陥 / 改善についての要約。
- **優先度:** バグ トラッキング システムで割り当てられた、欠陥 / 改善の相対的な重要性。
- **重要度:** バグ トラッキング システムで割り当てられた、欠陥 / 改善の相対的な影響度。
- **割り当て:** バグ トラッキング システムで割り当てられた、欠陥 / 改善のオーナー。
- **ステータス:** バグ トラッキング システム中の欠陥 / 改善のステータス。
- **Report Center ステータス:**
  - **未解決:** バグ トラッキング システムでまだ修正されていない。
  - **未テスト:** バグ トラッキング システムで修正済みだが、まだ Project Center によって登録されたテストがない。
  - **修正済み:** バグ トラッキング システムで修正済みである。Project Center によって登録されたすべてのテストが成功だった。
  - **失敗したテスト:** バグ トラッキング システムで修正済みだが、Project Center によって登録されたテストの一部が失敗だった。

**注意：**特定の欠陥 / 改善についてクリックされたすべてのテスト（自動および手動）を参照するには、ID をクリックして [欠陥 / 改善の編集] ページを開き、[自動テスト] タブに移動します。

- **テストが必要：**Project Center は、プロジェクトで修正された個々の欠陥をテストすることを要求します。欠陥をテストする必要がないと判断した場合、ユーザーは欠陥 / 改善を編集して設定を変更できます。ID をクリックし、[テストが必要] のチェックボックスをオフにします。
- **タスク：**欠陥 / 改善を修正するためにスケジュールされたタスク。プロジェクト中のどの欠陥 / 改善がすでにタスクとしてスケジュールされているかが分かるだけでなく、まだスケジュールされていない欠陥 / 改善も分かります。
  - ツールチップ：[タスク] カラムに表示される ID の上にマウス ポインターを置くと、関連するタスクのイテレーション ID とイテレーション名がツールチップに表示されます（図 25）。

図 25: イテレーション ID と名前のツールチップ

Project Center ステータス	テストが必要	タスク
	はい	111 [success]
	はい	7 [open]
	はい	5 [open]
	はい	

名前: Fix 欠陥 51 : National characters do not work in user profile data  
 ステータス: 完了  
 反復名: Milestone 1: Architecture design and core framework implementation  
 予定時間: 1d

# タスク レポート

以下のタスク レポートを利用できます。

- [タスクの統計] レポート
- [タスク アクティビティ] レポート

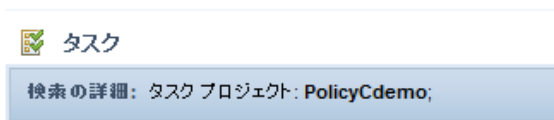
## [タスクの統計] レポート

[タスクの統計] レポートには、タスクの実装のステータスが表示されます。

検索条件バーを使って、表示する統計を指定します。検索条件バーの左側には、現在選択している検索条件が表示されます。右側には、「保存済み検索」のコンボ ボックスが表示されます。その隣のボタンを使って次の操作を行うことができます。

- 表示されたダイアログのフィールドを使ってタスクを検索する。
- 現在の検索条件を「保存済み検索」として保存する。
- 現在選択している「保存済み検索」を削除する。
- ページをリフレッシュする。

現在選択している検索条件は、以下の図のように検索条件バーの左側に表示されます。



統計は以下のグラフでレポートされます。

グラフ	説明
タイプごとのタスク -- 量	特定のタスク検索条件に対するタスク タイプの割合を示します。たとえば、開発、設計、管理、サポート、およびその他のアクティビティについて何個のタスクがあったかを確認できます。
タイプごとのタスク -- 実作業時間	各タスク タイプにかかった実作業時間の合計を示します。たとえば、開発、設計、管理、サポート、および他のアクティビティにチームがそれぞれ何時間かけたかを確認できます。
関係ごとのタスク -- 実作業時間	要求に関連するタスク、欠陥に関連するタスク、どちらにも関連しないタスクについて、実際にかかった作業時間の合計を示します。
テスト ステータスごとのタスク -- 量	各テスト ステータス (成功、失敗、未テスト、テスト不要) についてタスクの数を示します。

グラフ	説明
ステータスごとのタスク -- 量	各ステータス（オープン、作業中、完了、キャンセル）についてタスクの数を示します。
オーナーごとのタスク	特定のタスク検索条件について、特定の Development Testing Platform ユーザーに属するタスクの数を示します。
欠陥トップ 10 -- 実作業時間	修正するのに最も時間がかかった欠陥を示します。このグラフは、欠陥に関連付けられているタスクの実作業時間に基づきます。
改善トップ 10 -- 実作業時間	実装するのに最も時間がかかった改善を示します。このグラフは、改善に関連付けられているタスクの実作業時間に基づきます。
要求トップ 10 -- 実作業時間	実装するのに最も時間がかかった要求を示します。このグラフは、要求に関連付けられているタスクの実作業時間に基づきます。
タイプごとのタスク -- 計画された作業時間 vs 実作業時間	選択したタスク タイプについて、予定された作業時間と実際の作業時間を比較します。このグラフは、全体的な見積りの正確さを判断するのに役立ちます。
関係ごとのタスク -- 計画された作業時間 vs 実作業時間	選択したタスクの関係について、予定された作業時間と実際の作業時間を比較します。
オーナーごとのタスク -- 計画された作業時間 vs 実作業時間	選択したタスク オーナーについて、予定された作業時間と実際の作業時間を比較します。各チームメンバーが予想された時間中にタスクを完了しているかどうかを判断するのに役立ちます。
計画時間 -- ヒストグラム	X 軸は、タスクの計画時間の範囲を表します。棒グラフは、その範囲におけるタスクの数を示します。
実時間 -- ヒストグラム	X 軸は、タスクの実時間の範囲を表します。棒グラフは、その範囲におけるタスクの数を示します。

右上の [ 印刷可能なバージョン ] アイコンをクリックすると、印刷できるレポートが生成されます。

## [タスク アクティビティ] レポート

[タスク アクティビティ] レポートには、特定の期間にプロジェクトで作業されたタスクのリストが表示されます。このレポートから次の情報をすぐに見ることができます。

- プロジェクト チーム メンバーの現在のタスク
- 各チームメンバーが現在のタスクを終了するために必要な残り時間

- 指定の期間にチーム メンバーが作業したタスク
- チーム メンバーごとの平均タスク カバレッジ
- 中断されたタスクの数
- 作業進捗
- アクティビティのガントチャート

[ 選択されたプロジェクト メンバーのすべてのタスクを表示 ] チェックボックス ( 図 26) がオンの場合、[ タスク アクティビティ ] レポートには、選択したプロジェクトだけでなく、そのプロジェクトで作業しているメンバーの他のタスクもすべて表示されます。

図 26: 選択されたプロジェクト メンバーのすべてのタスクを表示

選択されたプロジェクト メンバーのすべてのタスクを表示 (タスクのプロジェクトに関係なく)

このチェックボックスがオフの場合は、選択したプロジェクトのすべてのタスクがそのオーナーと共に表示されます。

[ タスク アクティビティ ] レポートは次のセクションから構成されます。

- タスク アクティビティの概要
- タスク アクティビティの詳細

## タスク アクティビティの概要

タスク アクティビティの概要には、選択した期間中における各チーム メンバーのタスクについての統計が表示されます ( 図 27)。

図 27: タスク アクティビティの概要

タスクアクティビティの概要:

オーナー		完了	新規	アクティブ	中断	平均タスク数	カバレッジ	平均継続時間
<a href="#">JohnSmith</a>		0	0	1	0	1.000	100.0	48d 7h
<a href="#">admin</a>		0	0	1	0	1.000	100.0	74d 5h
<a href="#">kit</a>		0	0	1	0	1.000	100.0	78d 2h
<a href="#">kylie</a>		0	0	2	0	2.000	100.0	533d
<a href="#">masu</a>		0	0	3	0	3.000	100.0	539d
<a href="#">michael</a>		0	0	1	0	1.000	100.0	604d
<a href="#">agata</a>		0	0	3	0	3.000	100.0	504d
<a href="#">sew</a>		0	0	1	0	1.000	100.0	132d
平均		0.0	0.0	1.6	0.0	1.625	100.0	314d

チーム メンバーごとに以下の情報が表示されます。

- **完了** : 選択した期間にそのチーム メンバーが完了したタスクの数。
- **新規** : 選択した期間に、タスクのステータスが「作業中」に設定される前に、そのチーム メンバーに割り当てられたタスクの数。



- **アクティブ**: タスクのステータスが「作業中」に設定された後に、チーム メンバーが現在作業しているタスクの数。
- **中断**: チーム メンバーが作業していたが、完了前に中断しなければならなかったタスクの数。
- **平均タスク数**: そのチーム メンバーが同時に作業しているタスクの平均数。理想的な値は 1 です。
- **カバレッジ**: 選択した期間中に、そのチーム メンバーが「作業中」のタスクを少なくとも 1 個担当していた時間の割合。
- **平均継続時間**: 選択した期間中に、そのチーム メンバーがタスクの作業にかけた時間の平均。


適切なプロジェクト メンバーの隣の [アクティビティ グラフを表示] ボタン  をクリックすると、そのメンバーのアクティビティ グラフが表示されます。このガント チャート (図 28) は、選択された期間中にチーム メンバーが作業したタスクをグラフィカルに表現します。

図 28: タスク アクティビティ



## タスク アクティビティの詳細

タスク アクティビティの詳細セクションには、選択したプロジェクトで作業中のチーム メンバーごとに別の表が表示されます (図 29)。この表は、選択した期間にチーム メンバーが作業したタスクについて、その詳細な情報を提供します。

図 29: タスク アクティビティの詳細

kylie					現在の状態			
アクティビティ					現在の状態			
タスク ID	開始	終了	継続時間	ステータス	タスク名	反復	ステータス	見送り作業時間
33	2010-09-14	...	544d	作業中	要件 23 の実装: Req1		作業中	2d 0h
36	2010-10-13	...	523d	作業中	要件 27 の実装: 音声認識	8	作業中	5d 3h

masu					現在の状態			
アクティビティ					現在の状態			
タスク ID	開始	終了	継続時間	ステータス	タスク名	反復	ステータス	見送り作業時間
31	2010-09-09	...	547d	作業中	要件 20 の実装: 要件 1		作業中	0h 0m
32	2010-09-09	...	547d	作業中	変更要件 1	7	作業中	1d 0h
37	2010-10-13	...	523d	作業中	要件 28 の実装: 命令の実行 (走る、持つ、座る)	8	作業中	1d 0h

タスク アクティビティの詳細の各行は、タスクが「作業中」だったときのタイム フレームについての情報を提供します。そのため、2 度「作業中」だったタスクは、別々の行に 2 回表示されます。

タスク アクティビティの詳細の表は、終了日によってソートされます。つまり、最も直近に完了したタスクがリストの一番下になります。

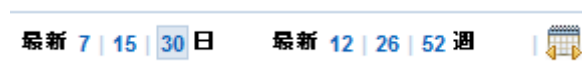
タスク アクティビティの詳細の表は以下の情報を提供します。

- **アクティビティ**：この部分には、特定のアクティビティの期間（開始から終了まで）の値が表示されます。
  - **タスク ID**：タスクに割り当てられた識別番号。
  - **開始**：タスクに対する作業が開始した日。言い換えると、タスクに「作業中」ステータスが設定されたときのタイムスタンプ。
  - **終了**：タスクの作業が完了した日。言い換えると、タスクのステータスが「作業中」以外に設定されたときのタイムスタンプ。
  - **継続時間**：選択された期間においてタスクの作業にかかった時間。
  - **ステータス**：特定の「作業中」期間の最後にチーム メンバーが設定したステータス。
    - **オープン**：選択された期間にチーム メンバーはこのタスクの作業を停止しなければならなかった。まだタスクは完了していない。
    - **完了**：タスクは正常に完了。
    - **キャンセル**：タスクはもう有効ではない。
    - **作業中**：チーム メンバーが現在作業中。
  - **現在の状態**：この部分には、現在の値（たとえば今日の状態）が表示されます。
    - **タスク名**：タスクの名前。タスクの名前をクリックすると、[ タスクの編集 ] ページが開き、タスクについての詳細情報が表示されます。
    - **イテレーション**：現在タスクが属しているイテレーション。
    - **ステータス**：現在のタスクのステータス。
    - **見残り残作業時間**：チーム メンバーがタスクを完了するのに必要な時間の推定（この見残り時間は [ タスクの編集 ] ページでチーム メンバーによって設定されます）。

## [タスク アクティビティ] レポートの期間の設定

[タスク アクティビティ] レポートでは、期間を指定してデータを表示することができます。リンクをクリックして、最近の 7 日、15 日、30 日、または最近の 12 週、26 週、52 週から期間を選択できます ( 図 30)。

図 30: 期間のリンク



今日の日付からリンクの日数 / 週の間データが表示されます。たとえば、今日が 8 月 7 日で "7 日" のリンクを選択した場合、8 月 1 日から 8 月 7 日の間に発生したアクティビティが表示されず。

## [タスク アクティビティ] レポートのデータ範囲の指定

データ範囲を設定することもできます。次の操作を行います。


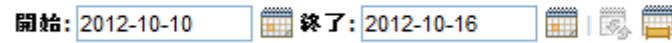

1. カレンダー ボタン  ボタンをクリックします。  
[開始] および [終了] フィールドが表示されます (図 31)。

図 31: 日付の範囲のフィールド



2. 日付をフィールドに入力するか、カレンダー ボタンを使って選択します。
3. [リフレッシュ] ボタン  をクリックして [タスク アクティビティ] レポートに新しいデータを表示します。

# イテレーションレポート

[イテレーションのステータス] レポートは、選択したプロジェクトのイテレーションとタスクについて以下の情報を提供します。

- イテレーションの詳細
- イテレーションのタスク

## イテレーションの詳細

選択したプロジェクト中のイテレーションごとに、[タスク ステータス] タブの下に次の情報が表示されます (図 32)。


- 予定されたスケジュール、現在のステータス、および作業期間の見積り
- 完了、キャンセル、および作業中のタスクの数。
- 予定されたスケジュールからの乖離。乖離がある場合、その日数。
- 予想されるイテレーションの最終日 (Project Center が予測する)

図 32: イテレーションの詳細

反復数: 4 タスク数: 53

ID	反復名	タスク数				予定		作業時間		見積り			
		合計	Comp.	Prog.	オープン	開始日	作業時間	終了日	現在のステータス		見積り		
									実作業時間	残作業時間	予定作業時間からの見積り乖離時間	終了日	終了予定日からの乖離
1	Milestone 3: Advanced functionality implementation and release	11	1	1	9	2012-09-26	60d	2012-10-26	50d 7h 54m	57d	47d 7h 54m	2012-11-21	18d
2	Milestone 2: Basic functionality implementation	12	0	0	12	2012-08-26	54d	2012-09-25		54d	0d 0h 0m	2012-11-09	33d
3	Milestone 1: Architecture design and core framework implementation	14	8	2	4	2012-07-26	54d	2012-08-25	1d	26d	-27d	2012-10-29	46d
4	フェーズ1	3	1	2	0	2010-10-08	29d	2010-10-20	1058d 4h 58m	6d 3h 58m	1036d 56m	2012-10-22	523d
5	[どの反復にも属していないタスク]	13	5	0	8				0d 0h 0m			2012-10-16	
合計		53	15	5	33		197d		1110d 4h 52m	143d 3h 58m	1057d 50m		620d

より具体的には、この表は以下の情報をイテレーションごとに提供します。

- **イテレーション ID:** 固有の識別番号。矢印ボタン  をクリックすると、対応するイテレーションタスクの表までスクロールします。
- **イテレーション名:** 固有の名前。イテレーション名をクリックすると、[イテレーションの編集] ページが開き、選択したイテレーションについての詳細情報が表示されます。
- **タスク数:**
  - **合計:** イテレーション中のすべてのタスクの数。
  - **Comp.:** イテレーション中の完了したタスクの数。
  - **Prog.:** イテレーション中の作業中のタスクの数。
  - **オープン:** オープンのタスクの数。

- **予定：**イテレーションの計画時での時間に関連する設定。
  - **開始日：**イテレーションの開始予定日。
  - **作業時間：**イテレーション全体を完了するために必要な作業日数の予測。この値は、イテレーション中のすべてのタスクの [ 予定作業時間 ] フィールドの合計です。
  - **終了日：**イテレーションの作業の終了予定日。イテレーションの最終期限。
- **現在のステータス：**
  - **実作業時間：**そのイテレーションの作業にかかった時間の合計。この値は、イテレーション中のすべてのタスクの [ 実作業時間 ] フィールドの合計です。
  - **予定からの乖離：**予定時間における変更。
- **見積り：**
  - **残作業時間：**イテレーションを完了するのに必要な見積り残作業時間の合計。この値は、イテレーション中のすべてのタスクの [ 残作業時間 ] フィールドの合計です。
  - **終了日：**予測されるイテレーションの作業最終日。この日付は Project Center によって予測されます。この日付は、最も作業量の多いプロジェクト タスクを担当しているチーム メンバーが、最期のタスクを完了する見込みの日付です。
  - **終了予定日からの乖離：**( 実作業時間 + 見積り残り時間 ) マイナス (-) 予定作業時間

## イテレーションのタスク

レポートの下部には、イテレーション タスクについての詳細が表示されます。イテレーションごとに次の情報が表示されます。

- タスク クリープ グラフ
- タスクの詳細

### タスク クリープ グラフ

タスク クリープ グラフ ( 図 33 および 図 34) は、イテレーションの期間内におけるタスク ステータスの傾向を表します。

図 33: タスク クリープ グラフ - 例 1

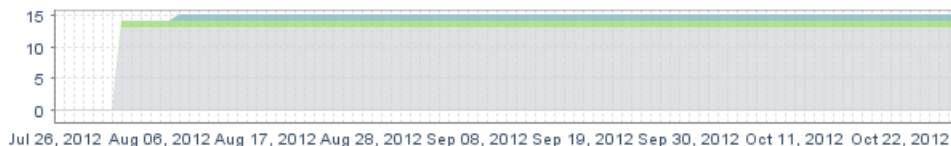
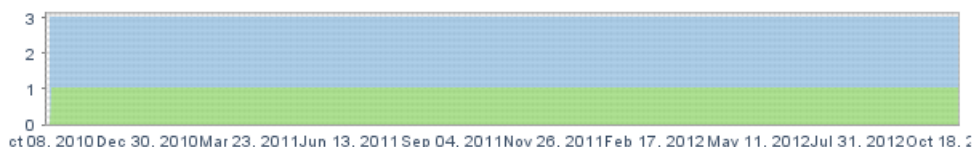


図 34: タスク クリープ グラフ - 例 2



各イテレーションのタスク クリープ グラフを見ると、プロジェクト マネージャーは以下の疑問に対する回答を得ることができます。

- イテレーションが予定または開始されてから、新しいタスクは追加されたか。追加された場合：
  - いつ新しいタスクが追加されたか
  - タスクのステータスにはどのような傾向があるか
- 完了したタスクには、どういった傾向があるか。イテレーションの最終期限に間に合わせることはできるか。
- 何個のタスクがキャンセルされたか。
- イテレーションの期間中に、何個のタスクが同時に作業中になったか。

## タスクの詳細

タスクの詳細は、対応するイテレーションのタスク クリープ グラフの下に表で示されます ( 図 35)。

図 35: タスクの詳細

ヘ ID	タスク名	ステータス	テストが必要	テストステータス	(再)テストが必要	リレーション	タグ	タスク所有者	予定作業時間	実作業時間	見積り残作業時間	予定からの見積り乖離
1	106 Architecture design - top level view	完了		不要				admin	5d	1d		-4d
2	107 Database structure design	完了		不要				JohnSmith	5d			
3	108 Easy user registration in the shop - server side	完了	✓	成功	🔴	REQ93		JohnSmith	4d			
4	109 Easy user registration in the shop - GUI front-end	完了	✓	未テスト		REQ93		JohnSmith	5d			
5	110 Forgot password functionality	完了	✓	失敗		REQ91		JohnSmith	4d			
6	111 Fix 次稿 51: National characters do not work in user profile data	完了		不要		PR51		DavidScott	1d			
7	112 User profile page - server side	完了	✓	成功		REQ94		DavidScott	4d			
8	113 User profile page - GUI front-end	作業中	✓	未テスト	🔴	REQ94		admin	5d			
9	114 Shopping cart functionality - add, edit, removed items - server side functionality	作業中	✓	未テスト		REQ101		JohnSmith	5d			

この表のカラムは次の情報を提供します。

- **タスク ID:** 各タスクの固有の ID 番号。
- **タスク名:** タスクの名前。タスク名をクリックすると、[ タスクの編集 ] ページが開き、タスクの詳細情報が表示されます。
- **ステータス:** チーム メンバーが設定した現在のタスクのステータス。以下のいずれかになります。
  - **オープン:** タスクはオープンであり、まだ完了していない。
  - **完了:** タスクは正常に完了した。
  - **キャンセル:** タスクは中断された。
  - **作業中:** タスクは現在作業中である。


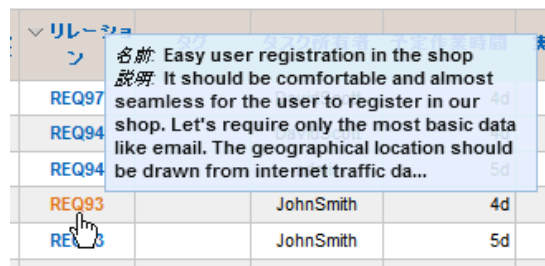
- **テストが必要**：このタスクにテストが必要かどうかを表します。これは、タスクを編集するときにチームメンバーが設定できます。
- **テストステータス**：タスクのステータスです。以下のいずれかになります。
  - **未テスト**：テストはまだ実行されていない。
  - **成功**：テストが成功し、タスクの品質が保証された。
  - **失敗**：テストが失敗した。
- **(再)テストが必要**：タスクがすでにテスト済み（ステータスは成功または失敗）であっても、何らかのテストを再実行するべきであると Project Center が判断した場合、そのタスクには丸い矢印のアイコン  が表示されます。そのような状況が起こるのは、たとえば、1 週間前にタスクがテストされていて、タスクに関連するソースファイルが 2 日前に修正された場合です。
- **リレーション**：タスクに関連する 欠陥 / 改善 または要求の ID が表示されます。欠陥 / 改善の背景色が紫の場合、その改善 / 欠陥はバグ トラッキング システムでクローズされています。このカラムが空の場合、タスクはスタンドアロン タイプです。
  - **ツールチップ**：[ リレーション ] カラムに表示される ID の上にマウス ポインターを置くと、関連する欠陥 / 改善または要求の概要とステータスがツールチップに表示されます ( 図 36)。

図 36: 概要とステータスのツールチップ



リレーション	名前	担当者	期限
REQ97	Easy user registration in the shop		4d
REQ94	It should be comfortable and almost seamless for the user to register in our shop. Let's require only the most basic data like email. The geographical location should be drawn from internet traffic da...		5d
REQ93		JohnSmith	4d
REQ94		JohnSmith	5d

- **タスク所有者**：タスクの所有者。
- **予定作業時間**：タスクを完了するために必要な作業日数。
- **実作業時間**：タスクの作業にかかった時間。
- **予定からの見積り乖離**：予定作業時間との乖離。
- **見積り残作業時間**：タスクを完了するために必要な残作業時間。

# テスト レポート

以下のテスト レポートを利用できます。

- [ テストの概要 ] レポート
- [ テスト セッション ] レポート
- 自動テストの詳細

さらに、関連する成果物の [自動テストの詳細](#) セクションで自動テストについての情報を得ることができます。

## [ テストの概要 ] レポート

[ テストの概要 ] レポートは、プロジェクト中のテスト、およびテストとプロジェクト成果物（要求、欠陥、タスク）との関係を表示します。このレポートは、プロジェクトの機能がどの程度良くテストされているか、および品質がどのレベルまで達成されているかを可視化します。

[ テストの概要 ] レポートは、シナリオの検索条件を変更することで、任意のシナリオ セットに対して生成することができます。デフォルトでは、プロジェクトのすべてのテスト シナリオが表示されます。

Test Overview

Search Details: Scenario project: SOAtest 9.2, Version: Latest, Show not run Scenarios: true. Saved Search: -- DEFAULT --

Scenario ID	22683
Scenario Name	Task 37937 - PR 90665 - Scripting editors: Evaluate button does not report the wrong number of arguments for methods not used as entry point
Scenario Description	Previously, the 'Evaluate' button was reporting argument count errors for methods not selected as the script entry point. While the script would still run with the selected entry point, the errors were misleading. Expected behavior: The error message for the wrong number of arguments should only apply to the method currently selected as the entry point.
Scenario Version	2
Criteria	
Test Session	6099 SOAtest 9.2 Acceptance Tests
Associated Task(s)	37937 Fix Def. 90665: CIBC - Scripting editors: Evaluate button reports wrong number of arguments for methods not used as entry point
Scenario Run	150126 Task 37937 - PR 90665 - Scripting editors: Evaluate button does not report the wrong number of arguments for methods not used as entry point - test 5
Scenario Run Start Date	2012-02-14 10:44:08
Scenario Run End Date	2012-02-14 11:07:18
Scenario Run Author	benken
Scenario Run Result	Passed

No.	Scenario Step	Expected result	Actual result	Result
1	Create a new empty .lst and test suite. Add an Extension Tool.			Passed
2	Create this script in Jython and save. Set the 'Method' to 'enterHere' and save.  from com.parasoft.api import Application  def enterHere(input, context): nonEntry(1, 2, 3) possibleEntry() Application.showMessage("enterHere: done")  def nonEntry(x, y, z): Application.showMessage("nonEntry: " + str(x) + ", " + str(y) + ", " + str(z))  def possibleEntry(): Application.showMessage("possibleEntry: ")			Passed

シナリオごとに以下の情報が表示されます。

- **シナリオの基本情報**：ID、名前、説明が含まれます。
- **シナリオのカスタム フィールド値**：シナリオでカスタム フィールドが定義されている場合、この情報が表示されます。



- **テストされた要求、欠陥、タスク**：これらの成果物にテストが関連付けられている場合、この情報が表示されます。
- **テスト シナリオの実行データ**：実行された日付と実行結果が含まれます。

レポートの表には、シナリオ ステップとその最後の実行結果が表示されます。

[ テストの概要 ] レポートは、MS Word の docx フォーマットにエクスポートしたり、印刷したり、保存することができます。この機能は特に認証監査 / コンプライアンス監査のために準備するときに役立ちます。

#### 注意

エクスポートされた .docx ファイル フォーマットは、Microsoft Office 2007 以降または Microsoft Word Viewer で開くことができます。.docx ファイル フォーマットは、Open Office または LibreOffice ではサポートされていません。

## [ テスト セッション ] レポート

[ テスト セッション ] レポートには、選択したプロジェクトでスケジューリングされたすべての手動テストセッションが表示されます。このレポートは以下の情報を提供します。

- セッションごとに分けられた特定のステータスのテスト数
- テスト セッションの予測開始日と実際の開始日
- 作業中または未完了のテスト セッションの最終期限
- テスト セッションが完了した日
- テスト セッションが実行された期間
- 各テスト セッションのステータス
- テスト結果によって決定されたセッションの最終ランク

[ テスト セッション ] レポートには、[ テスト ] > [ テスト セッション ] または [ レポート ] > [ テスト セッション ] からアクセスすることができます。

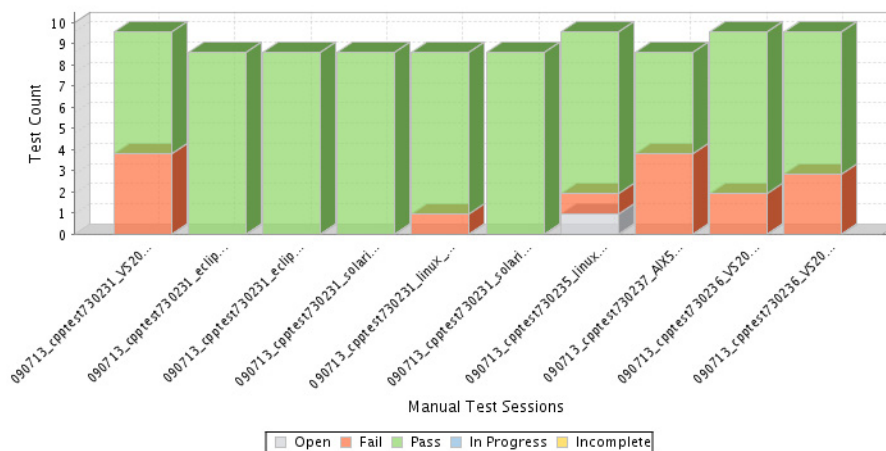
[ テスト セッション ] レポートは以下の 2 つのセクションから構成されます。

- テスト セッションのグラフ
- テスト セッションの詳細

### テスト セッションのグラフ

このグラフには、テストの数とそのステータスがセッションごとに表示されます ( 図 37 )。

図 37: 各セッションのテスト ステータス



## テスト セッションの詳細

各テスト セッションの詳細が表に表示されます。各テスト セッションで実行されたテストについてのデータです。

図 38: テスト セッションの詳細

ID	名前	ステータス	開始日	終了日	継続時間	反復 ID	開始予定日	最終期限	#未実行	#処理中	#失敗
<a href="#">2</a>	システムテスト	Ready to Run	2012-05-01	...		5	2012-05-01	2012-05-02	1	0	0
<a href="#">1</a>	機能テスト	Ready to Run	2012-04-30	...	2d	5	2012-05-01	2012-05-01	0	0	0
<a href="#">手動テスト セッションに所属していないシナリオ実行</a>										0	1
合計									1	0	1

テスト セッションの詳細の表には、以下の情報が表示されます。

- **ID:** 各テスト セッションの ID。ID をクリックすると、[手動テスト セッションの編集] ページが開き、テスト セッションの詳細を編集することができます。手動テスト セッションのスケジューリングと編集については 158 ページの「新しい手動テスト セッションのスケジューリング」を参照してください。
- **セッション名:** 各セッションの名前。
- **開始日:** テスト セッションが実際に開始した日。
- **終了日:** テスト セッションがクローズとしてマークされた日。
- **継続時間:** セッションが処理中か完了かによって、継続時間は次のいずれかを意味します。
  - **処理中:** これまでにテスト セッションが実行された時間の長さ。
  - **完了:** テスト セッションが完了するためにかかった時間の長さ。
- **イテレーション ID:** テスト セッションが属するイテレーションの識別番号。
- **開始予定日:** テスト セッションの開始予定日。

- **最終期限**：スケジュールされたテスト セッションの最終日。
- **# 未実行**：まだ実行されていない手動テストの数。
- **# 処理中**：処理中の手動テストの数。
- **# 失敗**：失敗した手動テストの数。
- **# 成功**：成功した手動テストの数。
- **# 未完了**：未完了の手動テストの数。
- **ランク**：手動テストの結果に基づいてテスト セッションに与えられた最終順位。

[手動テスト セッションに所属していないシナリオ実行] リンクは、テスト セッションの外部で実行されたすべての手動テストの実行を表示します。

## 自動テストの詳細

[自動テスト] タブには、特定のタスク、欠陥 / 改善、または要求に関連付けられたすべてのテストの最終実行が表示されます (図 39)。

図 39: [自動テスト] タブ



[自動テスト] タブは、[要求の編集] ページ、[欠陥 / 改善の編集] ページ、または [タスクの編集] ページにあり、成果物のタイプごとに詳細情報を表示します。

**注意**：自動テストとプロジェクト成果物を関連付けて [自動テスト] タブに表示されるようにする方法については、60 ページの「自動テストと Project Center との関連付け」を参照してください。

自動テストのセクションには、以下の情報が表示されます。

- 特定の要求、欠陥 / 改善、タスクに対して割り当てられて実行されたすべてのテスト ケースの詳細。
- すべてのテストの最終実行。テストの実行がどのくらい前かは関係しない。
- 最終実行の日付。
- 同じ名前が最終実行の日付が新しいテストを Development Testing Platform が受け取った場合、新しいテストの実行ステータスが表示されます。
- デフォルトでは、失敗したテスト ケースの詳細だけがツリー リストに表示されます。成功したテストの詳細も参照するには、[すべてのテストを表示] ボタンをクリックします。
- [プロジェクト] ドロップダウン メニューでプロジェクトを選択すると、[自動テスト] ビューがフィルターされ、選択されたプロジェクトに関連付けられたテストだけが表示されます。

### 自動テストの詳細

プロジェクト: \_\_\_\_\_

現在は失敗したテストの詳細だけを参照できます。すべてのテストの詳細を参照するには、[すべてのテ...

- ▼ **Functional Testing SOAtest@Pana-D1200-32** 0 1 10
- ▼ **Web\_API.tst** 成功 2012-09-27 16:02 0 1 10
- ▼ **Test 1: addNewItem** 失敗 2012-10-03 17:53 0 1 0
- ▼ **Test 1: addNewItem** 失敗 [TASK 626]

msgId	Value
status	failure
testcase_severity	Severity Level 1
line	1
test_category	Integration
error_type	1
testcase_name	Test 1: addNewItem
message	Connection reset POST /BookStoreSOAP/services/Car HTTP/1.0 Host: ██████████ Content-Type: text/xml; charset=UTF-8 Content-Length: 630 SOAPAction: ""

**注意：** 何らかのタスクまたは欠陥 / 改善が特定の要求と関連付けられていて、そのタスクまたは欠陥 / 改善に対して何らかのテストが実行された場合、実行結果は要求（[実行テスト] タブ）だけでなくタスクと欠陥 / 改善の両方に表示されます。

これらのプロジェクト成果物とテスト ケースをもう関連付けるべきではないと判断した場合、[Unassign This Test] ボタンをクリックしてこの要求、欠陥 / 改善、またはタスクとテスト ケースの関連付けを解除することができます。詳細については 68 ページの「テスト ケースと成果物の関連付けの解除」を参照してください。

# 付録 : Project Center

このセクションの内容 :

- Microsoft Project の統合
- MS Project アドインのインストール : トラブル シューティングとヒント
- タスク、欠陥、要求の電子メール通知
- バグ トラッキング システムとの同期
- 作業中のタスクの経過時間 ( 実時間 ) の計算方法

---

## Microsoft Project の統合

Microsoft Project プロジェクト マネージャーがタスクを計画する際によく使われるツールです。よく使われるのは、MS Project のデスクトップ アプリケーション版です。Parasoft DTP にもタスク管理機能があり、サーバー アプリケーションであるため、すべてのチーム メンバーが使用できます。たとえば、プロジェクト マネージャーが Parasoft DTP でタスクを計画し、開発者または QA チームのメンバーが各自のタスクを参照して作業の進捗を報告できます。

プロジェクト マネージャーは、MS Project アプリケーションと Parasoft DTP サーバーを統合し、2 つのシステム間で容易にタスクの情報を交換できます。これにより、Parasoft DTP と MS Project のプロジェクト計画の同期が維持されます。Parasoft DTP と MS Project の間で双方向の通信が可能です。

---

### ワークフローの例

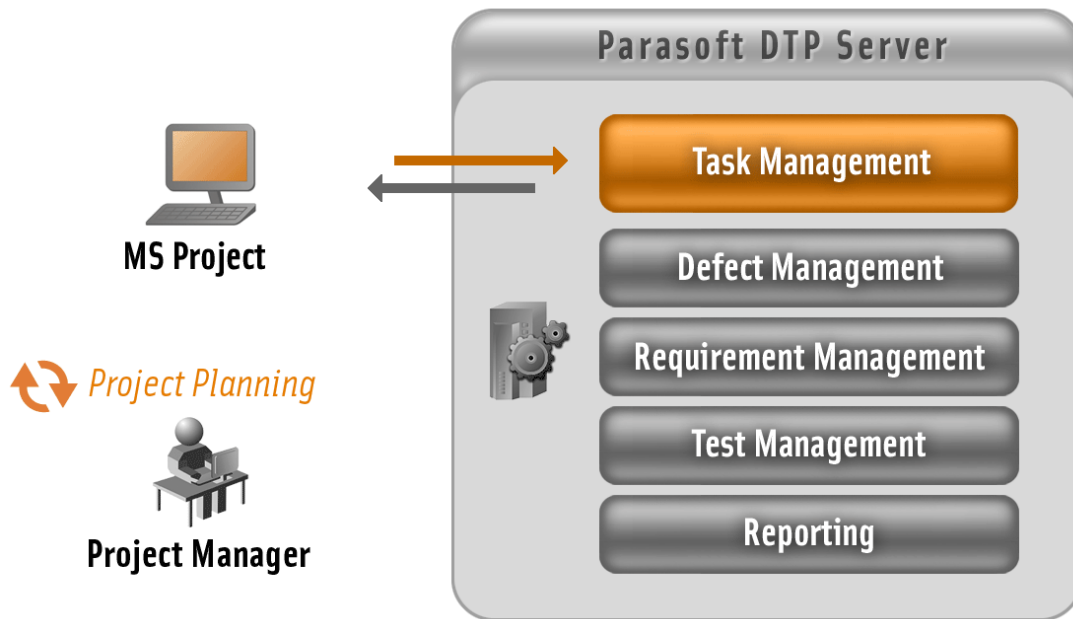
MS Project と Parasoft DTP を統合すると、たとえば以下のようなワークフローが可能になります。

---

#### MS Project で作業を計画し、計画されたタスクを Parasoft DTP Server にアップロードする

1. プロジェクト マネージャーは、MS Project のデスクトップ アプリケーションを使用して作業を計画します。複数のイテレーションやチームなどにわたる作業を計画できます。
2. プロジェクト マネージャーは、MS Project からプロジェクトを DTP サーバーに接続します。MS Project の既存のタスクごとに以下のアクションを実行できます。
  - Parasoft DTP サーバーの特定のイテレーションとタスクを関連付けることができます。

- タスクをローカルとしてマークし、Parasoft DTP に接続しないようにできます。

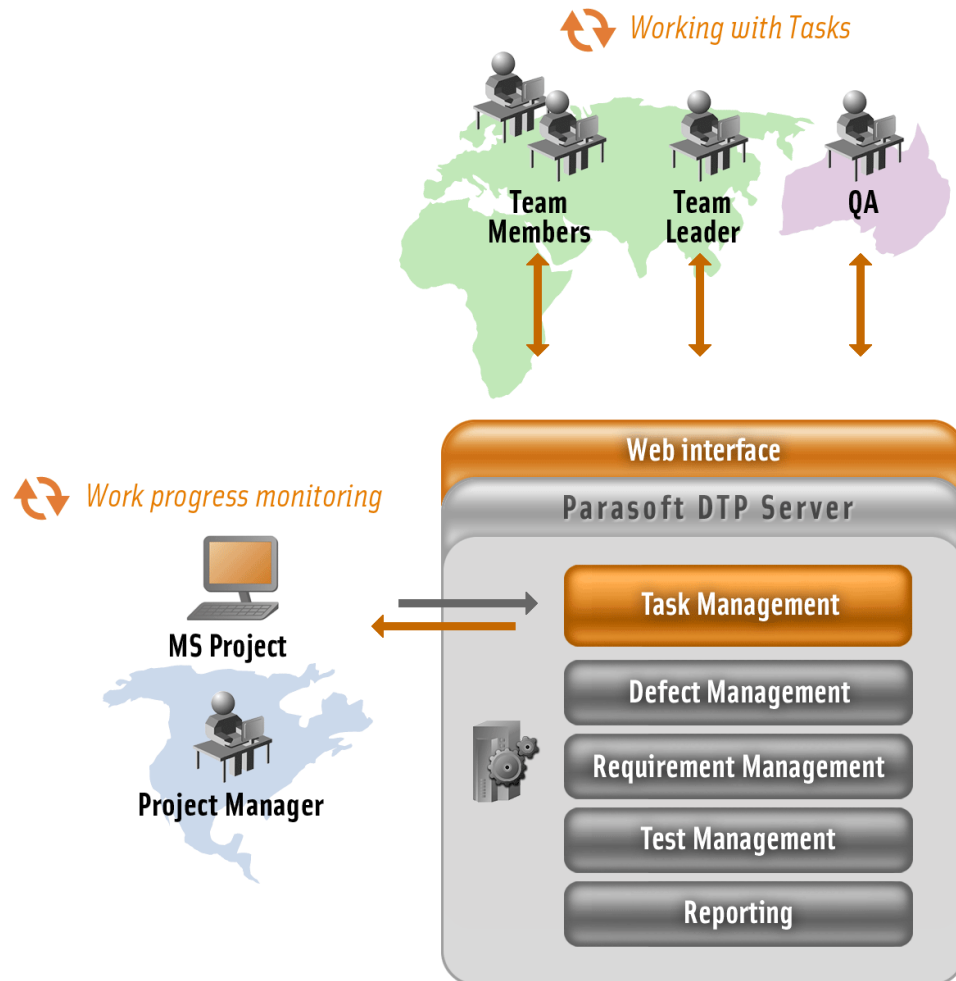


---

## Parasoft DTP でタスクを更新し、変更を MS Project にダウンロードする

1. プロジェクトまたはイテレーションの計画時にタスクを作成または変更します。
2. プロジェクトの実行中に、チームメンバーがタスクを作業します。チームメンバーはタスクのステータスを更新し、完了までの残り時間を設定します。

3. マネージャー / チーム リーダーは Parasoft DTP のレポートでイテレーションの進捗をモニターします。タスクのステータスを MS Project と同期できます。作業の進捗ステータスを MS Project で参照することもできます。



## ローカルな MS Project と Parasoft DTP Server の接続

Parasoft DTP と MS Project を統合するためのアドインが用意されています。MS Project から Parasoft DTP サーバーに接続してタスクを同期するには、MS Project にアドインをインストールします。

### 前提条件

MS Project にアクセスするには、以下のソフトウェアが実行中でなければなりません。

- 32 bit 版の MS Project 2003、MS Project 2007、または MS Project 2010
- 次のいずれかの Web ブラウザー



- Microsoft Internet Explorer  
Firefox と Microsoft ClickOnce アドオン \*  
Get ClickOnce for Firefox <https://addons.mozilla.org/en-us/firefox/addon/microsoft-net-framework-assist/>
- Chrome と ClickOnce for Google Chrome 拡張機能および Meta4 ClickOnce Launcher 拡張機能 \*  
これらの拡張機能の最新バージョンをダウンロードするには、Google Chrome Store を参照してください。  
\* DTP は、MS Project アドインのインストーラーを実行するときや、ウェブ ページから MS Project を開くときに、ClickOnce を使用してユーザビリティを改善します。Internet Explorer には、デフォルトで ClickOnce が組み込まれています。Firefox および Chrome は https: プロトコルでの ClickOnce 操作をサポートしていません。そのため、https: を使用する必要がある場合は、Internet Explorer を使用しなければなりません。
- Microsoft .NET Framework 3.5 (Windows 7 にはデフォルトで組み込まれています)

## MS Project アドインのインストール

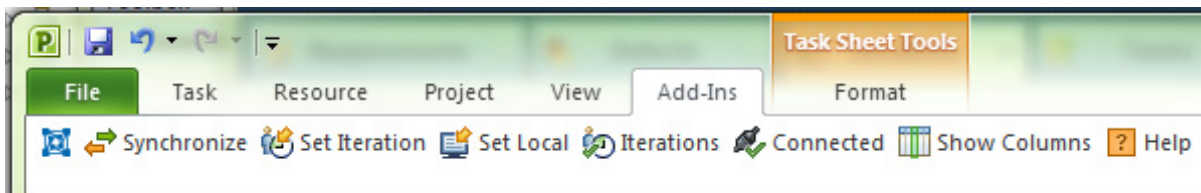
MS Project と DTP を接続するには、まず、Parasoft DTP に付属している MS Project アドインをインストールする必要があります。アドインは Parasoft DTP と MS Project を統合し、Parasoft DTP に接続してタスクを同期できるようにします。

1. Project Center ビューで [イテレーション] > [インポート] をクリックします。
2. [プラグインのインストールまたはアップデート] をクリックします。

ユーザーのマシンにアドインがダウンロードされ、ローカルな MS Project アプリケーションにインストールされます。

## MS Project から Parasoft DTP への接続

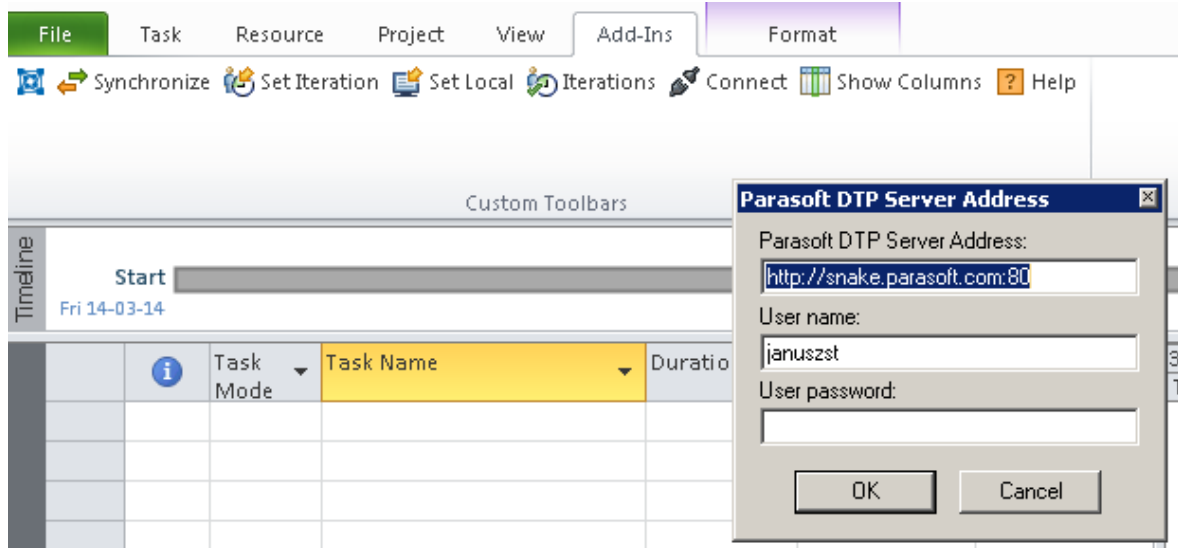
MS Project アドインをインストールすると、MS Project のツールバーに MS Project アドインのボタンが追加されます。これらのボタンは MS Project と Parasoft DTP の間でデータを交換するのを補助します。



### ツールバーがない場合

ツールバーがない場合は 218 ページの「MS Project アドインのインストール：トラブルシューティングとヒント」を参照してください。

1. [接続] をクリックし、Parasoft DTP サーバー接続の設定（ホスト、ユーザー名、パスワード）を入力します。



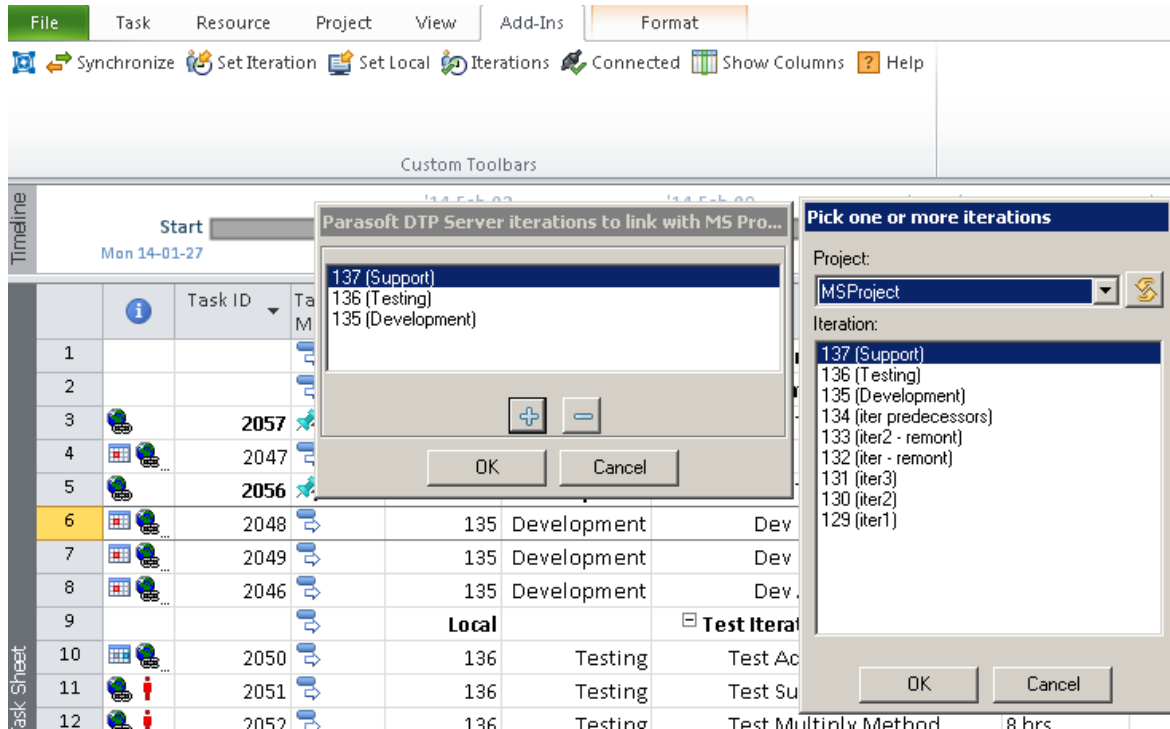
接続が確立されると、MS Project に次の列が追加されます。

- Parasoft DTP のタスクの値を表示する列：Task ID、Iteration ID、Iteration Name
- MS Project 固有の列：Actual Work および Remaining Work

	Task ID	Iteration ID	Iteration Name	Task Name	Actual Work	Remaining Work
	58436	2294	feedback processing	Fix PR 106330: Unexpected	3.22 hrs	8 hrs
	60598	2294	feedback processing	Fix Def. 108794 : Task ID a	0 hrs	8 hrs
✓	60599	2294	feedback processing	Help link redirects to nor	0.62 hrs	0 hrs
	58446	2294	feedback processing	Update MSProject integr	7.32 hrs	1 hr
✓	60594	2294	feedback processing	Fix Def. 108793 : It is not	9.55 hrs	0 hrs
✓	60618	2294	feedback processing	Fix Def. 108799 : Incompa	4.18 hrs	0 hrs

これらの列が表示されていない場合、[Show Columns] をクリックします。

- [Iteration] をクリックし、MS Project と統合する Parasoft DTP のイテレーションを選択します。



- [OK] をクリックし、ローカルな MS Project と選択された Parasoft DTP のイテレーションの接続を完了します。タスクはまだ Parasoft DTP からダウンロードされません。詳細については 213 ページの「SMS Project で接続されたタスクを Parasoft DTP と同期する」を参照してください。

## タスクの操作

Parasoft DTP のイテレーションまたは MS Project でタスクの追加、修正、削除を行うことができます。以下のタスク管理および同期操作を実行できます。

### 特定の MS Project のタスクと Parasoft DTP のイテレーションを関連付ける

- MS Project でタスクを選択します。
- MS Project のメインメニューで [Add-in] タブを選択し、[Set Iteration] をクリックします。

### MS Project のタスクをローカルなタスクとしてマークする

MS Project のタスクをローカルなタスクとしてマークし、Parasoft DTP とリンクしないようにできます。

- MS Project でタスクを選択します。

- MS Project のメインメニューで [Add-in] タブを選択し、[Set Local] をクリックします。イテレーションと関連付けられていないタスクは、[Iteration ID] 列に「Local」と表示されます。

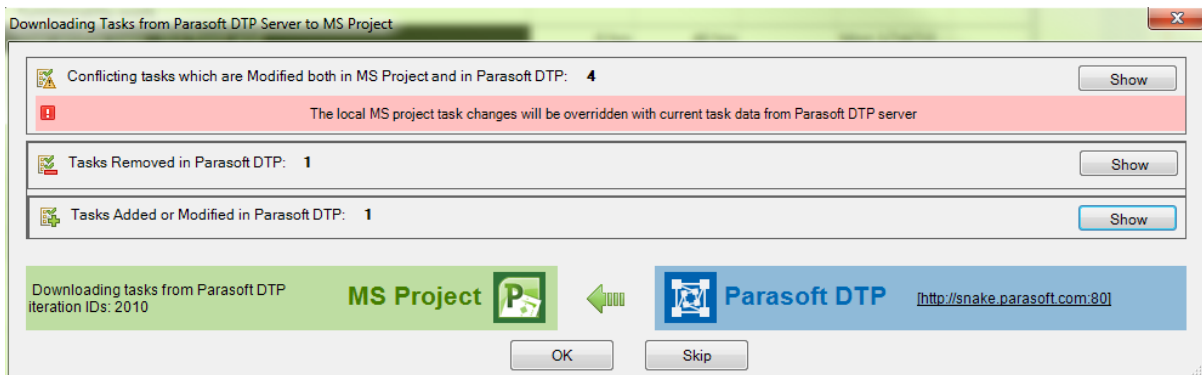
Start on 1/27/14							
Mon Jan 27      Tue Jan 28      Wed Jan 29      Thu Jan 30							
	Task ID	Iteration ID	Iteration Name	Task Name	Remaining Work	Actual Work	
		Local		Calculator Project	84 hrs	0.15 hrs	
		Local		Development Iteration	36 hrs	0.15 hrs	
	1	1	Development	Dev Add Method	10 hrs	0.15 hrs	
	2	1	Development	Dev Subtract Method	9 hrs	0 hrs	
	3	1	Development	Dev Multiply Method	9 hrs	0 hrs	
	4	1	Development	Dev Divide Method	8 hrs	0 hrs	
		Local		Test Iteration	32 hrs	0 hrs	
	5	2	Test	Test Add Method	8 hrs	0 hrs	
	6	2	Test	Test Subtract Method	8 hrs	0 hrs	
	7	2	Test	Test Multiply Method	8 hrs	0 hrs	
	8	2	Test	Test Division Method	8 hrs	0 hrs	
		Local		Support Iteration	16 hrs	0 hrs	
	9	3	Support	Support Task 1	8 hrs	0 hrs	
	10	3	Support	Support Task 2	8 hrs	0 hrs	

### SMS Project で接続されたタスクを Parasoft DTP と同期する

MS Project と Parasoft DTP の間でタスクを同期すると、Parasoft DTP Server から MS Project にタスクがダウンロードされるか、MS Project から DTP Server にタスクがアップロードされます。

#### Parasoft DTP Server から MS Project へのタスクのダウンロード

- タスクを選択し、[Synchronize] をクリックします。
- ダウンロードすべき差異または変更がある場合、次のダイアログが表示されます。

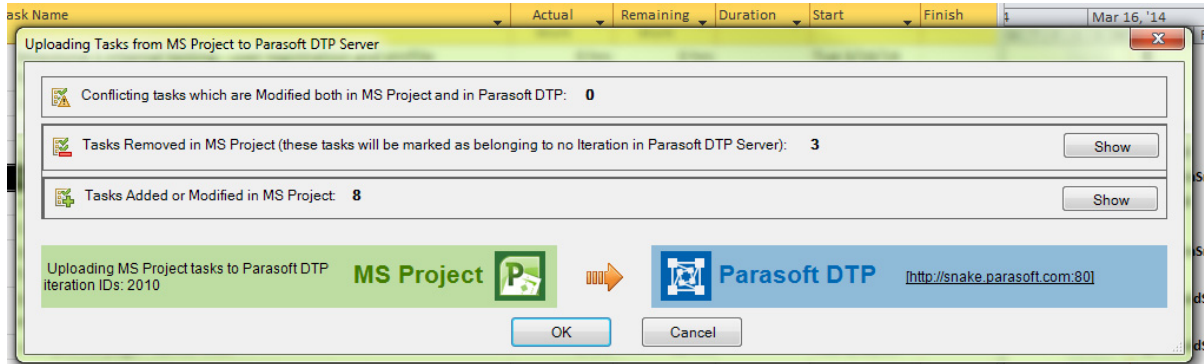


- 各セクションの [Show] をクリックすると、検出された競合によって MS Project のタスクに発生する削除、更新、上書きの詳細を参照できます。

4. [OK] をクリックし、Parasoft DTP からの変更を MS Project に適用します。

### MS Project から DTP へのタスクのアップロード

1. タスクを選択し、[Synchronize] をクリックします。
2. アップロードすべき差異または変更がある場合、次のダイアログが表示されます。



3. 各セクションの [Show] をクリックすると、検出された競合によって DTP のタスクに発生する削除、更新、上書きの詳細を参照できます。
4. [OK] をクリックし、MS Project からの変更を Parasoft DTP Server に適用します。

Project Center の GUI を使用して DTP Server にタスクをインポートすることもできます。詳細については 216 ページの「DTP から MS Project のタスクを Parasoft DTP Server にアップロードする」を参照してください。

## MS Project と Parasoft DTP のタスクのフィールドのマッピング

Parasoft DTP と MS Project のタスクのフィールドは似ていますが、同一ではありません。次の表は、タスクの情報を同期する際にどのフィールドが使用されるかを表しています。

Parasoft DTP のタスクのフィールド	MS Project のタスクのフィールド	注意
Remaining Time	Remaining Work	Remaining Time、Actual Time、および Start Date フィールドは、子タスク（リーフ タスク）のないタスクの場合にだけ同期されます。 親タスクの時間情報は、MS Project と Parasoft DTP の間で同期されません。
Actual Time	Actual Work	これは、MS Project と Parasoft DTP では、親タスクの時間の扱いが異なるからです。MS Project では、Actual Work または Remaining Work は、子タスクの値の合計です。DTP では、親タスクの情報は子タスクとは独立しています。
Start Date	Start	Actual Work、Remaining Work および Start Date フィールドを MS Project で変更し、Parasoft DTP にアップロードできるのは、タスクがまだ作業開始されていない場合、つまり、Actual Work = 0 の場合だけです。
Planned Time	Actual Work + Remaining Work	Planned Time フィールドに Actual Work と Remaining Work の合計が設定されるのは、タスクがまだ作業開始されていない場合、つまり、Actual Work = 0 の場合だけです。 同期は一方向にだけ行われます。Parasoft DTP の Planned Time に基づいて MS Project のフィールドが更新されることはありません。
Task Name	Name	
Owner	Resource	
Parent	Parent	Parasoft DTP または MS Project でタスクの親子階層または先行のタスクが変更された場合、変更された関係が同期され、他方のシステムも最新の状態が保たれます。
Predecessors	Predecessors	MS Project と同期されているイテレーションに子タスクがあり、親タスクは MS Project と同期されていない別のイテレーションにある場合、子タスクがルートとして表示されます。

Actual Time および Remaining Time フィールドは、それぞれ Actual Work および Remaining Work フィールドにマップされ、Duration および Percent Completed フィールドにはマップされません。MS Project では、Actual Work および Remaining Work 列はデフォルトで表示されず、MS Project を DTP server に接続した後に初めて利用できるようになります。

## 典型的な使用シナリオの例

次の例では、MS Project で開発プロジェクトが計画され、リーフ タスクが Parasoft DTP サーバーにリンクされています (Task ID、Iteration ID 列に注目してください)。タスクは、Parasoft DTP の 3 つのイテレーションにスケジュールされています。最上位のサマリー タスクはローカルとしてマークされているため、Parasoft DTP サーバーとはリンクされません (Iteration ID 列の Local という文字列に注目してください)。

Start on 1/27/14						
Mon Jan 27		Tue Jan 28		Wed Jan 29		Thu Jan 30
Task ID	Iteration ID	Iteration Name	Task Name	Remaining Work	Actual Work	
	Local		Calculator Project	84 hrs	0.15 hrs	
	Local		Development Iteration	36 hrs	0.15 hrs	
	1	1	Development Dev Add Method	10 hrs	0.15 hrs	
	2	1	Development Dev Subtract Method	9 hrs	0 hrs	
	3	1	Development Dev Multiply Method	9 hrs	0 hrs	
	4	1	Development Dev Divide Method	8 hrs	0 hrs	
	Local		Test Iteration	32 hrs	0 hrs	
	5	2	Test Test Add Method	8 hrs	0 hrs	
	6	2	Test Test Subtract Method	8 hrs	0 hrs	
	7	2	Test Test Multiply Method	8 hrs	0 hrs	
	8	2	Test Test Division Method	8 hrs	0 hrs	
	Local		Support Iteration	16 hrs	0 hrs	
	9	3	Support Support Task 1	8 hrs	0 hrs	
	10	3	Support Support Task 2	8 hrs	0 hrs	

## DTP から MS Project のタスクを Parasoft DTP Server にアップロードする

1. Project Center で [ イテレーション ] > [ インポート ] を選択します。
2. 画面上の操作指示に従います。





## Import Iteration from MS Project

### 1. Install or Update Parasoft Development Testing Platform Plugin in your MS Project

Parasoft provides plugin for MS Project which allows synchronization between MS Project tasks and Parasoft Development Testing Platform tasks. You can use MS Project to work with Development Testing Platform tasks (track their progress or reschedule them) and then commit the changes back into Concerto. Both MS Project 2003, 2007 and 2010 are supported.

[Install or Update Plugin](#)

### 2. Open your MS Project File

Launch MS Project on your computer and open your MS Project file with tasks you plan to bind with Parasoft Development Testing Platform Iteration.

### 3. Choose Target Iteration

Decide whether you would like bind your tasks to an existing iteration or create a new iteration. If you would like import into a new iteration in Development Testing Platform, click create it.

[+ New Iteration](#)

### 4. Bind your MS Project Tasks with Parasoft Development Testing Platform Iteration

In MS Project, navigate to Parasoft Development Testing Platform add-in. Connect to Development Testing Platform server by pressing [Connect] button. Associate the selected MS project tasks with Development Testing Platform target iterations by pressing [Set Iteration] button. Once your MS Project tasks are associated with the specific Development Testing Platform iterations, you can fully synchronize your MS Project file with Development Testing Platform server by pressing [Synchronize] button on MS Project Development Testing Platform add-in.

アップロードされるタスクが、Parasoft DTP のアカウントをまだ持っていないオーナーに関連付けられている場合、インポート時にアカウントが作成されます。このアカウントのデフォルトのパスワードは `developer` です。

---

## その他のヒントおよび注意事項

- MS Project で親タスクにタスク オーナーを割り当ててはいけません。MS Project の サマリー タスクは、子タスクの作業の合計を表しています。
- MS Project または Parasoft DTP で親ではないタスクには、タスク オーナーを 1 人だけ割り当ててください。
- 親タスクの時間情報 (Actual Time、Remaining Time、Start Date) は、MS Project と Parasoft DTP の間で同期されません。これは、MS Project と Parasoft DTP では、親タスクの時間の扱いが異なるからです。MS Project のサマリータスクでは、Actual Work または Remaining Work は、子タスクの値の合計です。DTP では、親タスクの情報は子タスクとは独立しています。



# MS Project アドインのインストール：トラブルシューティングとヒント

このセクションでは、MS Project アドインのインストールでよくある質問を取り上げます。MS Project アドインをインストールする方法については 207 ページの「Microsoft Project の統合」を参照してください。

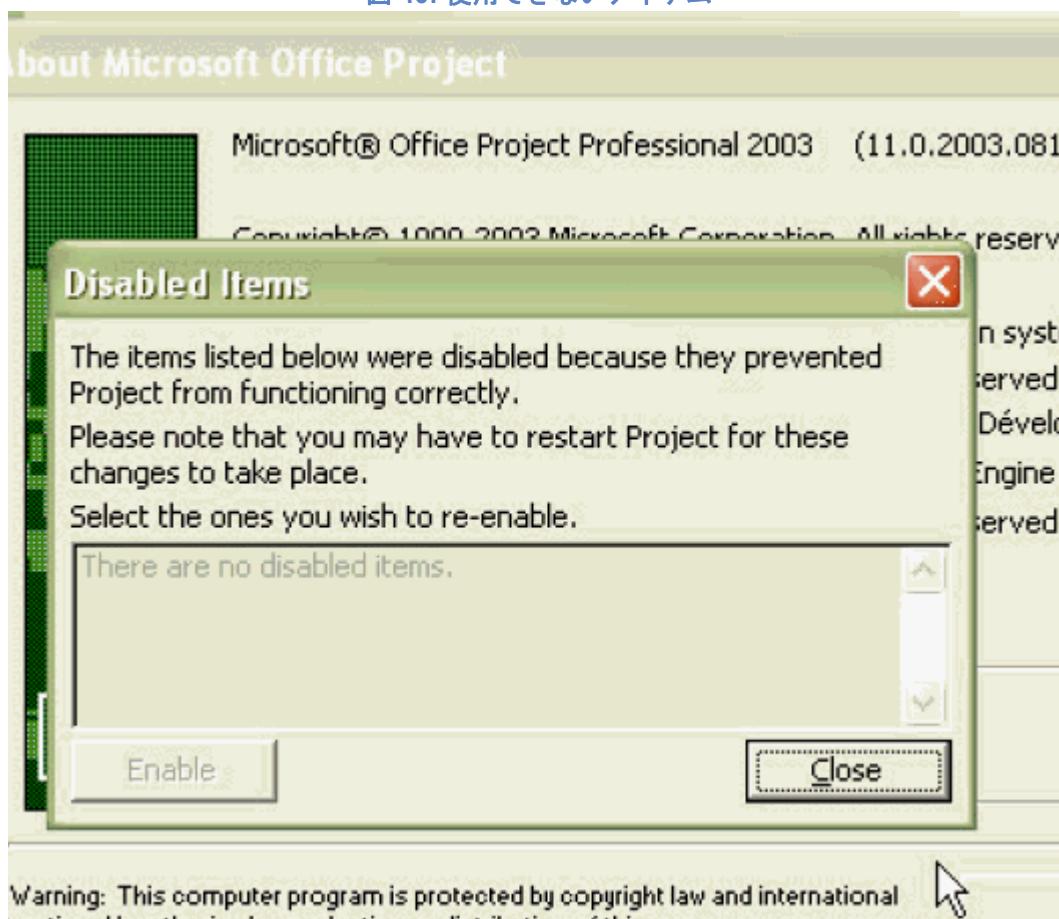
## トラブルシューティング

1. **質問：** MS Project アドインのインストールは問題なかったが、MS Project を実行したときにツールバーにこのアドインが表示されない。

**回答：**

- このアドインがデフォルトで無効になっている可能性があります。MS Project 2003 の場合、[ ヘルプ ] > [ Microsoft Office Project について ] をクリックし、[ 使用できないアイテム ] ボタンをクリックしてください ( 図 40 )。

図 40: 使用できないアイテム



- Component Object Model (COM) アドインをロードまたはアンロードし、操作指示に従います。操作が終了すると、アドインのリストに表示される Development Testing Platform MS Project アドイン (com.parasoft.pts.planning.tasks.msproject.addin) のチェックボックスがオンになるはずですが (図 42)。チェックボックスがオフの場合、チェックボックスをオンにして MS Project を再起動します。
- Microsoft Office 2003/2007/2010 Primary Interop Assemblies が正しくインストールされなかった可能性があります。これは [コントロール パネル] の [プログラムと機能] からチェックできます (適切なバージョンの Microsoft Office Primary Interop Assemblies がインストールされているかどうかを確認します)。リストにない場合 (図 42)、次の Web ページから手動でインストールしてください。
  - MS Office 2003: <http://www.microsoft.com/downloads/en/details.aspx?FamilyID=3C9A983A-AC14-4125-8BA0-D36D67E0F4AD>
  - MS Office 2007: <http://www.microsoft.com/downloads/en/details.aspx?FamilyID=59DAEBAA-BED4-4282-A28C-B864D8BFA513>
  - MS Office 2010: <http://www.microsoft.com/downloads/en/confirmation.aspx?FamilyID=938fe8ad-583b-4bd7-a345-23250dc15855>

図 41: Microsoft Primary Interop Assemblies

プログラムのアンインストールまたは変更

プログラムをアンインストールするには、一覧からプログラムを選択して [アンインストール]、[変更]、または [修復] をクリックします

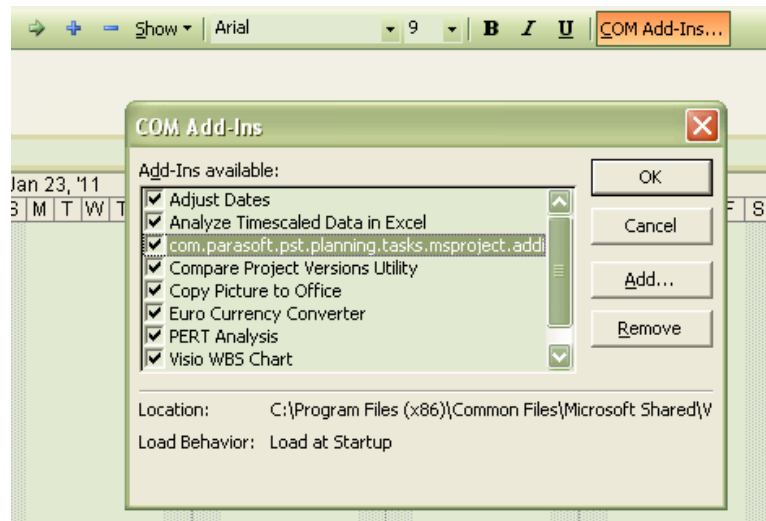
名前	発行元	インストー...	サイズ	ノ
Microsoft Office 2010 Primary Interop Assemblies	Microsoft Corporation	2012/10/19	6.85 MB	1
Microsoft Office Project Professional 2007	Microsoft Corporation	2012/10/10		1

別の方法として、これらのファイルは Development Testing Platform サーバーのディレクトリにも格納されています。

DTP\_HOME\tomcat\webapps\grs\msproject\addin\

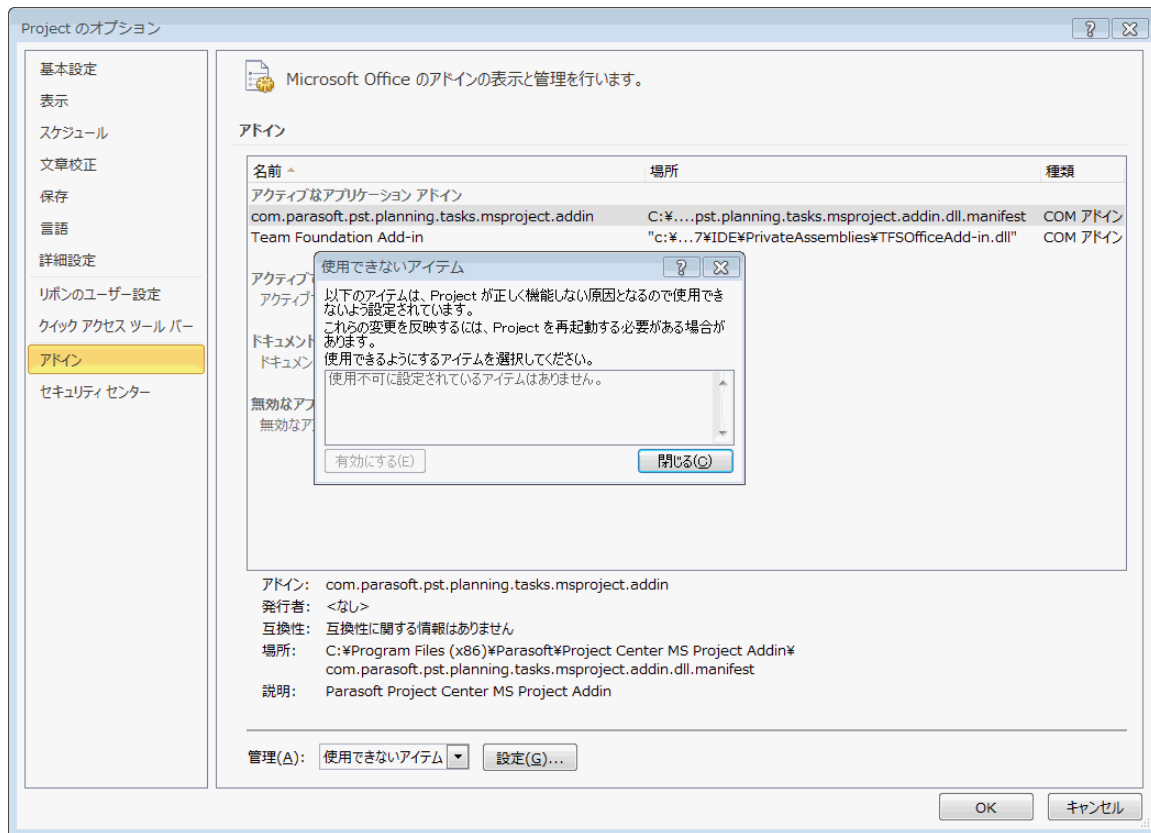
使用している MS Project のバージョンに従って、o2003pia.msi、o2007pia.msi、o2010pia.msi のいずれかのファイルを実行します。MS Project を閉じてからもう一度開きます。MS Project アドインがツールバーに表示されるはずですが。

図 42: COM アドイン



- MS Project 2010 の場合、[ファイル] > [オプション] をクリックして [アドイン] を選択し、[Manage] コンボボックスから [使用できないアイテム] を選択します。

図 43: 使用できないアイテム



com.parasoft.pst.planning.tasks.msproject.addin がリスト中にあるかどうかを確認します。リスト中にある場合、有効にして MS Project を再起動します。

2. **質問：** MS Project アドインで問題が発生した場合、Development Testing Platform のサポートに送るログ ファイルはどこにあるのか？

**回答：** ログ ファイルは、現行ユーザーの Windows アプリケーション データ フォルダの ParasoftProjectCenterMSProjectAddin フォルダにあります。たとえば、以下の Windows のパスでログ ファイルを発見できます。

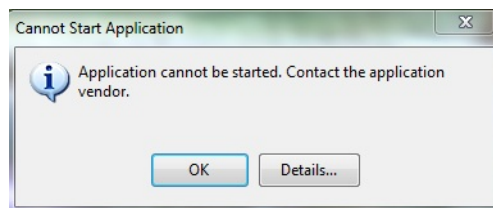
- Vista:  
c:\Users\[USER\_LOGIN]\AppData\Local\ParasoftProjectCenterMSProjectAddin\
- XP:  
c:\DocumentsandSettings\[USER\_LOGIN]\LocalSettings\Application Data\ParasoftProjectCenterMSProjectAddin.

3. **質問：** アドインをアップグレードしたのだが、MS Project にアドインが表示されなくなった。

**回答：** 一部の古いデータが Windows でキャッシュされた可能性があります。下記の操作を行ってください。

1. ユーザー設定の Apps フォルダに移動します。たとえば Windows Vista では c:\Users\[USER\_LOGIN]\AppData\Local\Apps\ です。Windows XP では、c:\Documents and Settings\[USER\_LOGIN]\Local Settings\Apps です。
2. このフォルダの内容をバックアップします。
3. このフォルダの内容を削除します。
4. Development Testing Platform MS Project アドインをアンインストールしてから再びインストールします。

4. **質問：** Windows 7 を使用している。Development Testing Platform の Web ページから MS Project アドインをインストールしようとしたが、「Application cannot be started. Contact the application vendor.」というメッセージ ダイアログが表示される。



**回答：** Windows 7 のユーザーアクセス制御メカニズムが高レベルに設定されていて、Web ブラウザーが MS Project アドインの実行を制限されている可能性があります。これを防ぐには、まず Web ブラウザーを右クリックして [Run as Administrator] を選択します。そして Development Testing Platform の Web ページに移動して MS Project プラグインをインストールします。

5. **質問：** MS Project アドインをインストールするときに、「Application cannot be started. Contact the application vendor.」というエラー メッセージが表示される。ログには以下の情報が記録される。

```

+ Store metadata "AppType" is not valid.
+ Input string was not in a correct format.

--- Inner Exception ---
System.FormatException
- Input string was not in a correct format.
- Source: mscorlib
- Stack trace:
at System.Number.StringToNumber(String str, NumberStyles options,
NumberBuffer& number,
NumberFormatInfo info, Boolean parseDecimal)
at System.Number.ParseUInt32(String value, NumberStyles options,
NumberFormatInfo numfmt)
at System.UInt16.Parse(String s, NumberStyles style, NumberFormatInfo
info)
at System.Convert.ToUInt16(String value)
at
System.Deployment.Application.ComponentStore.GetPropertyAppType(Defin
itionAppId appId, String propName)

```

**回答:** 次の Windows レジストリ エントリの値が不正なことが原因です。

HKEY\_CURRENT\_USER\Control Panel\International\sPositiveSign Windows Support

詳細については <http://support.microsoft.com/kb/942460> を参照してください。問題のあるキーを直接編集することもできますが、推奨する方法は、場所の設定を別の地域と言語に変更してから元の設定に戻すことです。次の操作を行います。

- a. コントロール パネルの [ 地域と言語 ] をクリックします。
- b. 現在の形式をメモ帳などに控えます。
- c. 現在の [ 形式 ] を別の形式に変更します。たとえば [ 英語 ( オーストラリア ) ] などに変更します。
- d. [ Apply ] をクリックします。
- e. 現在の [ 形式 ] をステップ b で控えた形式に設定します。
- f. [ Apply ] をクリックして [ OK ] をクリックします。このアクションは、HKEY\_CURRENT\_USER\Control Panel\International レジストリ キーの下の個々の値をリセットします。
- g. マシンを再起動します。

# タスク、欠陥、要求の電子メール通知

Project Center は、タスク、欠陥、または要求が作成または変更されたときに、すべての関係者に電子メールによる通知を送信します。

## 変更ベースの通知

変更ベースの通知電子メールは、タスク、欠陥、要求が変更されたときに、以下のユーザーに送信されます。

- タスク、欠陥、要求のオーナー。オーナーが変更された場合は、以前のオーナーと現在のオーナーの両方に通知が送信されます。
- タスク、欠陥、要求の監督者。任意のユーザーを「監督者」として追加し、タスクの E-mail 通知を受け取るようにできます。監督者はいつでもタスクに追加できます。詳細については 123 ページの「タスクの新規作成」を参照してください。
- 要求の作成者。詳細については 123 ページの「タスクの新規作成」を参照してください。
- プロジェクトのチーム リーダー (オプション)。プロジェクトのチーム リーダーが電子メール通知を受け取るには、チーム リーダーのプロファイル ページでオプションを有効にする必要があります。過剰な電子メールの送信を避けるため、チーム リーダーへの通知は、デフォルトではオフに設定されています。

### User Profile

変更ベースの通知電子メールは、変更をコミットしたユーザーには送信されません。たとえば、ユーザー A がタスクを作成した後にタスクを修正しても、変更についての E-mail は A には送信されません。タスクのオーナー / 監督者が A 以外のユーザーである場合、それらのユーザーに E-mail が送信されます。

新しく作成したタスクの場合、タスクに関するすべてのデータが通知に含まれます。

## タスクの E-mail 通知 - 新規タスク

PARASOFT Concerto			
実行ユーザー admin on 2012-10-19 16:36			
 <b>[TASK 108] SOAtestで単体テスト (WSDLベース)</b>			
関係			
テストが必要	false		
プロジェクト	MyProj1	Iteration ID	9
優先度	最高		
オーナー	masuya	監視者	nagata
名前	SOAtestで単体テスト (WSDLベース)		
ステータス	オープン		
説明			
予定	0d 0h 0m		
ストーリー ポイント			

タスクが修正された場合、変更されたタスクに関連する情報が通知されます。

## タスクの E-mail - 変更されたタスク

PARASOFT Concerto		
実行ユーザー admin on 2012-10-19 16:37		
 <b>[TASK 108] SOAtestで単体テスト (WSDLベース)</b>		
変更日時	開始	終了
予定作業時間	0d 0h 0m	2d
開始予定日		2012-10-22


E メール通知の設定の詳細については 272 ページの「タスクと欠陥の通知メールの設定」を参照してください。

## タスクの日次ダイジェスト通知

日次ダイジェストは毎日午前零時過ぎに生成され、前日に特定のプロジェクトで行われたすべてのタスクの変更が記載されます。タスクの日次ダイジェストは、プロジェクトのチーム リーダーに送信されます。プロジェクト チーム リーダーの指定方法については、230 ページの「プロジェクトの定義」を参照してください。

その日のタスクに対する変更をまとめたメールが毎日午前零時過ぎに送信されます。このメールは特定のプロジェクト リーダー（プロジェクトのマネージャーとして定義されたユーザー）に送信されます。

タスクの日次ダイジェスト E-mail のサンプル

<b>Project:</b>		<b>Concerto</b>			
Task Activity in your project:		<a href="#">View report</a>			
<b>New Tasks</b>					
ID	Name	Created by	Owner(s)	Planned	
<a href="#">44899</a>	Req Specification Report - click on req ID opens in iframe	januszst	pawelf	1h	
<a href="#">44900</a>	Fix Def. 117543 : ClassCastException when modifying Parent custom field	pawelf	pawelf	1d	
<a href="#">44906</a>	Fix Def. 117536 : QA: Test Sessions report shows incorrect number of not run scenarios.	jeehongm	jeehongm	1d	
<a href="#">44907</a>	Support for Iter 962	mtran	mtran	3d	
<b>Reopened Tasks</b>					
[No task has been reopened]					
<b>Closed Tasks</b>					
ID	Name	Closed by	Owner(s)	Planned	Actual
<a href="#">44888</a>	Fix some l10n issues in Japanese version	pawelf	pawelf	1d	1d 4h 23m
	 New Comment fixed.				
<a href="#">44900</a>	Fix Def. 117543 : ClassCastException when modifying Parent custom field	pawelf	pawelf	1d	3h 38m
	 New Comment fixed.				
	There was a few problems: <ol style="list-style-type: none"> <li>1. Custom field definition did not handle value type other then String (Parent ID is int).</li> <li>2. Not all placed did handle multiple values (hierarchy tasks)</li> </ol> All issues should be fixed now.				
<a href="#">44118</a>	2 issues on new Chrome	pawelf	pawelf	4h	
	 New Comment tested on chrome version: 21.0.1180.89 m				



## バグ トラッキング システムとの同期

Development Testing Platform は一部のバグ トラッキング システム (BTS) と統合します。この統合によって、93 ページの「Development Testing Platform による欠陥管理の容易化」で説明する機能を利用できます。

サポートされるバージョンの Bugzilla、HP Quality Center、または JIRA と Development Testing Platform が統合している場合、94 ページの「タスクと欠陥の同期化」の説明にあるように、ステータスの変更とコメントが Development Testing Platform からそのバグ トラッキング システム中の関連する課題に伝播されます。

たとえば、以下のようにこの統合を利用できます。

1. Bugzilla の欠陥を Development Testing Platform のタスクとしてチーム リーダーがスケジューリングし、開発者に割り当てます。
2. この開発者が Development Testing Platform でタスクに対する作業を開始し、タスクを「作業中」としてマークし、[バグ トラッキング システムを更新] オプションを有効にしてこのタスクを保存した場合、Development Testing Platform は Bugzilla 中の関連する欠陥のステータスを自動的に "assigned" に変更します。
3. Development Testing Platform のタスクにコメントを開発者が追加し、[バグ トラッキング システムを更新] オプションを有効にしてこのタスクを保存した場合、Development Testing Platform はこのコメントを Bugzilla の欠陥にも追加します。


 バグ トラッキング システムの更新

4. Development Testing Platform でタスクが完了し、[バグ トラッキング システムを更新] オプションを有効にしてこのタスクを保存すると、Development Testing Platform は Bugzilla の欠陥を「fixed/resolved」に設定します。

## 管理者向けの情報

- サポートされるバグ トラッキング システムのバージョンについて、およびバグ トラッキング システムを Development Testing Platform と統合する方法については、336 ページの「バグ トラッキング システムおよび要求管理システムの統合」を参照してください。
- この更新を設定する方法については 347 ページの「BTS Updater の構成」を参照してください。

# 作業中のタスクの経過時間 (実時間) の計算方法

タスクに対して作業する間、タスクのステータスは「作業中」にするべきです。ステータスを「作業中」にすると、Project Center は各ユーザーのタスク作業時間のモニタリングを開始します。この作業時間はタスクの [ 実時間 ] フィールドに表示されます。このタスクのステータスが「作業中」から別のステータスに変わると、作業期間の終了日が設定されます。

モニタリングされた作業期間はそれぞれ [ タスクの編集 ] ページの [ アクティビティ ] の表に 1 行で表示されます。

## アクティビティ

1 アイテム			
ユーザー名	開始日	終了日	継続時間
kylie	2010-10-13 17:26		523d 7h 12m

各タスクにかかった作業時間の合計は、[ アクティビティ ] の表のすべての「継続時間」の合計として計算されます ( 特定のタスクの「開始日」から「終了日」までの時間として計算されるものではありません )。

タスクの「作業中」の継続時間が計算されると、Project Center はデフォルトの作業時間を月曜日から金曜日の 8 AM から 4 PM と仮定するとともに Development Testing Platform サーバーの時刻を考慮します。同じ日に 8 時間を超えたアクティビティは「1 日」として扱われません。

**注意：** Development Testing Platform サーバーの時刻と世界各地との間には時差があります。将来は特に時間帯の違いについて、ユーザーごとにこの設定がカスタマイズされる予定です。

この計算アルゴリズムは以下のように動作します。

- 「作業中」ステータスが 1 日間続く場合 ( つまり開始時間と終了時間がどちらも同じ日の 8 AM から 4 PM の間である場合 )、予測される残作業時間は実時間に一致します。
  - 対象となる日が週末の場合、その日は平日として計算されます ( たとえば、あるチームメンバーが土曜日または日曜日に出社してタスクのステータスを「作業中」に変更し、同じ日にタスクを「作業中」から別のステータスにした場合です )。計算される作業時間は、そのタスクに費やされた実時間に一致します。
- 「作業中」ステータスが数日間続く場合：
  - 初日は開始時刻から 4 PM までです。開始時刻が 4 PM より後の場合、予測される残作業時間の継続時間はゼロです。
  - 経過日は 8 時間続きます。  
( 作業時間は、この日に開始せず、終了しません。 )
  - 最終日は 8 AM から終了時刻までです。終了時刻が 8 AM よりも前の場合、その日の継続時間はゼロです。

**注意：** 経過日が週末 ( 土曜日または日曜日 ) の場合、経過日は計算されません。

3. 例 :

h.

- 開始 : 月曜日 , 11 AM
- 終了 : 月曜日 , 3 PM
- 計算された継続時間 : 4h

i.

- 開始 : 月曜日 , 11 AM
- 終了 : 火曜日 , 3 PM
- 計算された継続時間 : 5h 月曜日 + 7h 火曜日 = 12h = 1d 4h (就業日は 8 時間続く)

j.

- 開始 : 月曜日 , 11 AM
- 終了 : 水曜日 , 3 PM
- 計算された継続時間 : 5h 月曜日 + 8h 火曜日 + 7h 水曜日 = 20h = 2d 4h

k.

- 開始 : 月曜日 , 10 AM
- 終了 : 月曜日 , 8 PM
- 計算された継続時間 : 10h = 1d 2h  
ユーザーは 1 日 (8 時間) よりも長く作業したが、経過日をフルに使用しなかった。

l.

- 開始 : 月曜日 , 6 AM
- 終了 : 火曜日 , 8 PM
- 計算された継続時間 :  
月曜日 6 AM - 4 PM (デフォルトの終了時刻) = 10h  
火曜日 8 AM (デフォルトの開始時刻) - 8 PM = 12h  
10h + 12h = 22h = 2d 6h

m.

- 開始 : 月曜日 , 1 AM
- 終了 : 月曜日 , 3 AM
- 計算された継続時間 : ユーザーは就業時間外で作業を開始して終了したが、同じ日であるため経過時間は 2h である。

---

## 管理者

- DTP のインストール
- Development Testing Platform の設定
- Report Center および Project Center とサードパーティ製ツールの統合
- 付録 : Development Testing Platform 管理者ガイド

## プロジェクトの作成と構成

Project Center と Report Center でのほとんどのアクティビティは、特定のプロジェクトに関連して実行されます。DTP ユーザー マニュアルの「プロジェクトの作成および設定」章では、プロジェクトに最も関係の深いアクションが説明されています。マニュアルを参照するには、DTP の UI で [ヘルプ] リンクをクリックしてください。

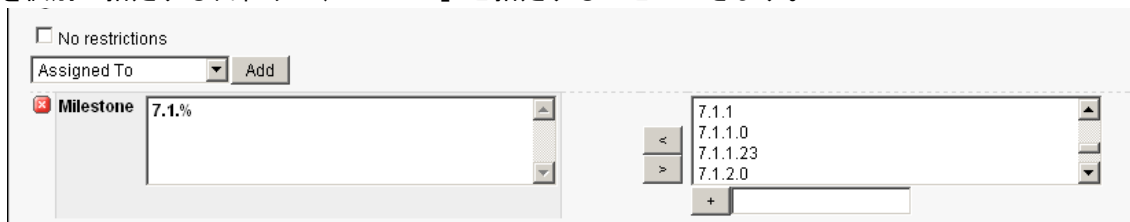
### プロジェクトの定義

フィルターなどのプロジェクト パラメーターを定義して、プロジェクトに関連付けられたデータを管理することができます。デフォルトでは、すべてのプロジェクトの全データに対してチームメンバーのアクセスが制限されます。ただし、プロジェクト マネージャーは必要に応じてプロジェクト フィルターを定義して、特定のプロジェクトへのアクセスをメンバーに許可できます。また、プロジェクトの定義で [制限なし] オプションが有効な場合、すべてのデータが利用可能になります。ただし、プロジェクト マネージャーはこのオプションを無効にしてデータに制限をかけることができます。

#### プロジェクト定義フィルターでのワイルドカードの使用

ワイルドカードとしてパーセント記号 (%) を使って、任意の文字または文字列を検索することができます。例を挙げて説明します。

たとえば、「7.1.1, 7.1.1.0, 7.1.1.23, 7.1.2.0」などのように C++test 7.1 のマイナーバージョンを個別に指定する代わりに、「7.1.%」と指定することができます。



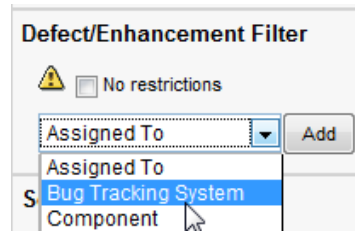
C++test 7.1 のすべてのマイルストーンが検索されます。

### 欠陥 / 改善フィルター

336 ページの「バグ トラッキング システムおよび要求管理システムの統合」で説明しているように、Project Center は数多くのバグ トラッキング システムと統合します。

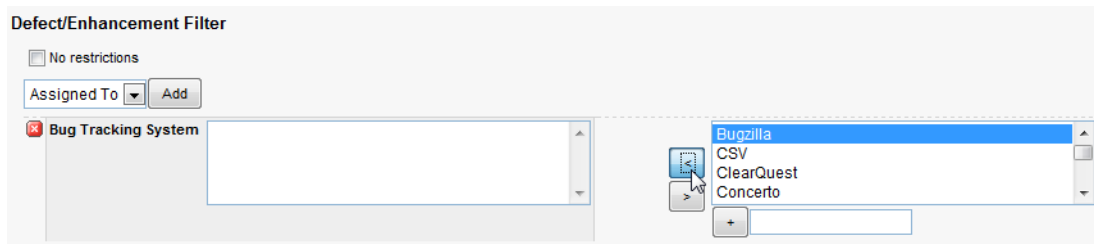
プロジェクトとバグ トラッキング システムを関連付けるには、次の操作を行います。

1. [欠陥/改善フィルター] のドロップダウン リストから [バグ トラッキング システム] を選択して [追加] ボタンをクリックします。



Development Testing Platform (DTP) に統合されたバグ トラッキング システムの一覧が表示されます。このリストには、DTP の管理者が各システムに割り当てたラベルが使用されます。

2. 右側のシステムの一覧から、統合するバグ トラッキング システム (BTS) を選択し、[<] ボタンをクリックして BTS をプロジェクトと関連付けます。

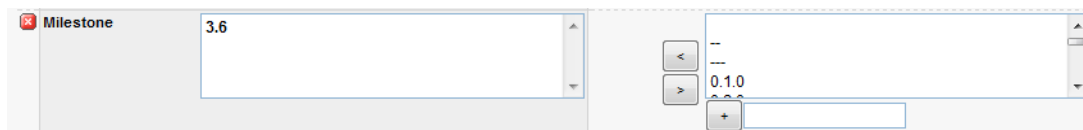


選択したバグ トラッキング システムのすべての項目がプロジェクトと統合されます。

特定のプロジェクトだけにフィルター条件を制限するには、次の操作を行います。

1. [欠陥/改善フィルター] のドロップダウン リストから [プロジェクト] を選択します。
2. [追加] ボタンをクリックします。これらは、統合されたバグ トラッキング システム からインポートされたプロジェクトです。
3. 右側のプロジェクトの一覧から、統合するコンポーネントを選択し、[<] ボタンをクリックしてこのコンポーネントをプロジェクトに関連付けます。

この方法で (つまり [欠陥/改善フィルター] のドロップダウン リストからフィルター カテゴリを選択し、左側のリストに目的のフィルター値を追加することで)、フィルターをさらに細かく設定できます。



## ログ フィルター

ログ フィルターを使用すると、Report Center に送信されたログに基づいて制限を設定することができます。たとえば、Jtest から送信されたデータだけを表示するように Report Center を設定でき

まず、Report Center に結果を送信するツールはすべてログを作成します。ログはツールの実行についての記録です。ログには次の定義済みプロパティがあります。

- **マシン** ツールを実行したシステムがあるマシンの名前。
- **OS のアーキテクチャ** ツールを実行したシステムのアーキテクチャ。
- **OS の名前** ツールを実行したシステムがあるオペレーティング システムの名前。
- **OS のバージョン** ツールを実行したシステムがあるオペレーティング システムのバージョン。
- **ツール** テストを実行したツールの名前またはシンボル。たとえば Jtest 8.0 がテストを実行した場合、ツールは「Jtest」になります。
- **ツールのバージョン** テストを実行したツールのバージョン番号。たとえば Jtest 8.0 がテストを実行した場合、バージョンは「8.0」になります。
- **ユーザー** システムがツールを実行したときにログインしていたユーザーのアカウント。

上記のいずれかのログ フィルターに基づいて、データを制限するフィルターを作成するには、次の操作を行います。

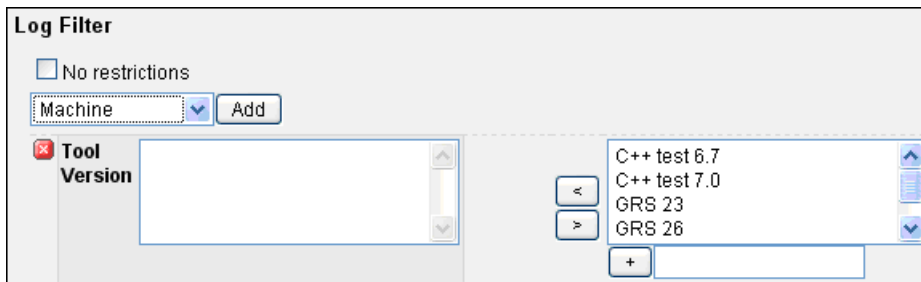
1. [ ログ フィルター ] で [ 制限なし ] チェックボックスをオフにします。

### プロジェクトフィルターの定義



2. [ ログ フィルター ] ドロップダウン リストから項目を選択し、その隣の [ 追加 ] ボタンをクリックします。

### ログフィルター



3. 制限を指定します。

右側のリストには、すでに Report Center に送られたプロパティのすべての値が表示されます。一部の値はインストール時に Report Center データベースに追加されます。そのため、まだ一度もツールを実行していない場合でもツールの値がリストに表示されます。

- **ログによる制限を設定するには** 右側のリストから適切な項目を選択し、[<] ボタンをクリックします。選択した項目が左側のリストに移動します。左側のリストにある項目によってデータが制限されます。
- **制限を削除するには** 左側のリストから項目を選択し、[>] ボタンをクリックします。選択した項目が右側のリストに戻ります。
- **Report Center データベースにまだない値 (したがってリストに表示されない値) を使って制限を設定するには** 右側のリストの下にあるテキスト フィールドに値の名前を入力し、[+] ボタンをクリックします。入力した値がリストに追加されます。

**注意:** ワイルドカード (%) を使って検索を拡張することができます。詳細については 230 ページの「プロジェクト定義フィルターでのワイルドカードの使用」を参照してください。

4. 構成を保存します。

## ログ プロパティ フィルター

各ログは、固定のプロパティだけでなく属性を持つことができます。ログ プロパティ フィルターを使用すると、属性に基づいてデータをフィルタリングできます。

ログ プロパティ フィルターを設定するには、次の操作を行います。

1. [ログ プロパティ フィルター] の [制限なし] チェックボックスをオフにします。
2. ドロップダウン リストから適切なログ プロパティを選択し、[追加] ボタンをクリックします。  
追加したいログ プロパティがない場合、次のステップに進んでください。

### ログ プロパティ フィルター

The screenshot shows a window titled "Log Properties Filter". Inside, there is a text input field containing the word "Build", followed by a downward-pointing arrow icon. To the right of this field is an "Add" button. Further to the right is another text input field, and to its right is an "Add New" button.

3. 次のいずれかのタスクを実行します。
  - **新しいログ プロパティをリストに追加するには:** [新規追加] ボタンの隣のフィールドにログ プロパティの名前を入力して [新規追加] ボタンをクリックします。入力したログ プロパティが表示されます。
  - **複数のログ プロパティを設定するには:** ログ プロパティは必要なだけ設定することができます。プロジェクトでは、1 つのログにつき 1 つの構成だけが存在することができます。
  - **ログ プロパティを削除するには:** 削除するログ プロパティの隣の赤い [+] ボタンをクリックします。
4. [プロジェクト フィルターの定義] タブの一番下の [保存] ボタンをクリックします。



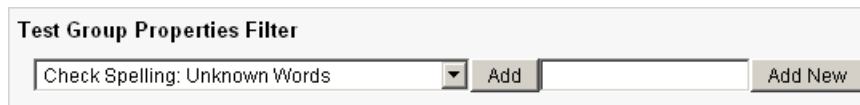
## テスト グループ プロパティ フィルター

テスト グループ属性を利用してデータをフィルタリングすることができます。テスト グループ プロパティ フィルターは、ログ プロパティ フィルターと同じように動作します。異なる点は、テスト グループ プロパティ フィルターの属性はログではなくテスト グループに割り当てられることです。

Report Center は数多くのテストを処理します。テスト結果を分かりやすくまとめるために、テスト結果は「テスト グループ」に分類されます。テスト グループは一連のテストと他のテスト グループから構成されます。ログ フィルターの場合と同じように、各テスト グループは関連する属性を持ちます。テスト グループの場合、属性はテスト実行全体ではなく特定のテスト グループを表します。たとえば、静的解析テストのテスト グループは静的解析に関連する属性を持ちます。ホワイト ボックス テストのテスト グループはホワイト ボックス テストに関連する属性を持ちます。

テスト グループ プロパティ フィルターを利用すると、Report Center のレポート処理のためのプロジェクトをより柔軟に定義することができます。テスト グループ プロパティ フィルターの適用方法はログ プロパティ フィルターの場合と同じです。

### テスト グループ プロパティ フィルター



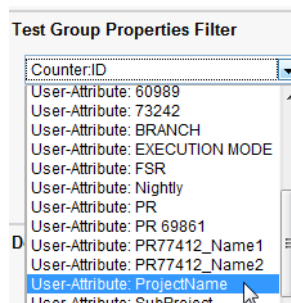
### 例

フィルターを作成して、DTP が受け取る自動テスト結果の中でどのテスト結果を特定のプロジェクトに関連付けるかを定義できます。

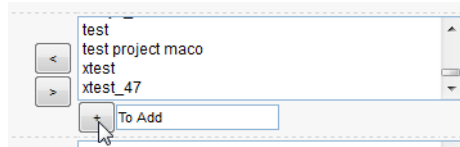
そのようなフィルターを定義する前に、Parasoft Test ツールが DTP に結果を必ず送信するようになる必要があります。詳細については 329 ページの「Parasoft Test から Report Center/Project Center へのテスト結果の送信」を参照してください。この設定が完了したら、Parasoft Test ツール (Jtest、C++test、dotTEST など) によって実行される自動テストは "User-Attribute: *ProjectName*" でマークされた結果を送信します。

関係のある自動テストをプロジェクトに関連付けるよう Project Center を構成するには、次の操作を行います。

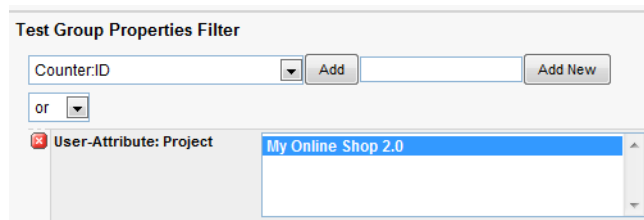
1. [ テスト グループ プロパティ フィルター ] ドロップダウン リストから [User-Attribute: *ProjectName*] を選択します。



2. [追加] をクリックします。以下のプロジェクトが表示されます。
  - a) 検索条件に一致している、および
  - b) DTP にテスト結果がある。
    - 目的のプロジェクト名がリストにない場合、そのような属性でマークされたテスト結果を DTP がまだ受け取っていないことを意味します。名前を入力して [ + ] ボタンを押して、そのような属性を手動で追加することができます。



3. 適切なプロジェクト名を選択し、[ < ] ボタンをクリックして左側のリストにプロジェクトを追加します。



ここまでの操作が完了すると、自動テストの結果が Parasoft Test ツールから DTP に送られて、ユーザー指定の属性（この例では "User-Attribute: ProjectKey=JBank1015"）でテスト結果がマークされている場合、そのテスト結果はプロジェクトに関連付けられます。

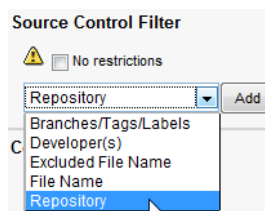
## ソース管理フィルター

392 ページの「Report Center とソース管理システムの統合」で説明しているように、Project Center は複数のソース管理システムと統合します。

ソース管理システムと DTP を統合する場合、ソース管理システムは Parasoft Source Scanner によって定期的にスキャンされ、プロジェクトコードのリビジョン情報が Project Center データベースに渡されます。

Project Center とソース管理システムを統合するには、次の操作を行います。

1. [ソース管理フィルター] のドロップダウン リストから [リポジトリ] を選択します。



2. [追加] ボタンをクリックします。リポジトリの一覧 (DTP の管理者によって指定された Source Scanner プロジェクトの名前) が表示されます。

3. 右側の一覧から、統合するリポジトリの名前を選択して [ < ] ボタンをクリックし、プロジェクトのソース管理フィルターの一部としてこのリポジトリを追加します。

ソース管理フィルターをさらに制限するには、このように条件を追加していきます。たとえば、DTP プロジェクトに該当する特定のソースパスだけを指定するには、次の操作を行います。

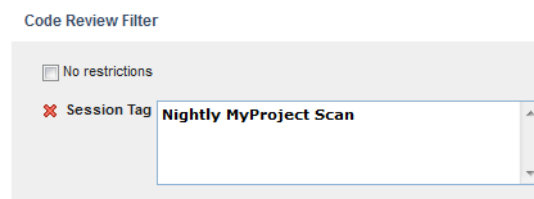
1. [ ソース管理フィルター ] のドロップダウン リストから [ ファイル名 ] を選択します。
2. 値のフィールドで、関連するソースコードへのパスを指定します。
  - 指定したファイル名で開始するソースコードへのすべてのパスがプロジェクトに関連付けられます。
  - ワイルドカードとしてパーセント記号を使用できます。
3. [ 追加 ] ボタンをクリックします。ファイル名の一覧が表示されます。
4. ファイル名の一覧から、プロジェクトに関連付けるファイル名を選択します。
5. [ < ] ボタンをクリックして、プロジェクトに関連付けるファイル名としてこれを追加します。
6. [ 保存 ] ボタンをクリックしてプロジェクトの構成を保存して終了します。

既存のデータにこのフィルターを適用する場合、[ 再計算 ] ボタンをクリックします。

## コード レビュー フィルター

コード レビュー フィルターは、Report Center のコード レビュー レポートに表示される結果をフィルタリングします。コード レビュー フィルターを定義する方法は、ログ フィルターの場合と似ています (231 ページの「ログ フィルター」を参照)。

### コード レビュー フィルター



コード レビュー セッション タグはカスタム文字列です。コード レビューの結果に紐付けられ、コード レビュー データ パックを区別するために使用されます。このセッション タグは、Parasoft Test の [Code Review] テスト コンフィギュレーションのセッション タグと一致するように設定すべきです (386 ページの「プロジェクトのコード レビュー レポートの設定」を参照)。

セッション タグを指定するほかに、次のようにフィルター制限をコントロールすることもできます。

- [ 制限なし ] がオンの場合：  
コード レビュー レポートには、選択されたプロジェクト チーム メンバーのすべてのコード レビューの結果が表示されます (どのプロジェクトやソース ファイルに対してコード レビューを実行したかは関係ありません)。
- [ 制限なし ] がオフであり、セッション タグが定義されて設定されている場合：  
DTP プロジェクトの設定で定義されたプロジェクト チーム メンバーが少なくとも一人コード レビューに関わっている限り、コード レビュー レポートには、選択されたセッション タグ

グでマークされたすべてのコード レビューの結果が表示されます。また、たとえ指定のセッション タグによる結果中にコード レビュー データが存在する場合でも、プロジェクト チームのリストにないユーザーのコード レビューは表示されません、詳細については 386 ページの「プロジェクトのコード レビュー レポートの設定」を参照してください。

## Parasoft Test の設定

「現行プロジェクト」のテストを実行するために Parasoft Test 製品 (C++test、dotTEST、SOAtest、Jtest) を使用するとき、このページで設定した localsettings が使用されます。

Parasoft Test ツールごとにファイルを作成する代わりに、このページで一度に localsettings を設定できます。この設定は、Parasoft Test 製品が DTP プロジェクトに接続するとき、自動的にその Parasoft Test 製品に伝播されます。

例：

```
#Report Center Settings
grs.data.port=32323
#License Settings
license.network.host=ntp.company.com
license.network.port=2222
license.use_network=true
#Mail Settings
report.mail.domain=company.com
report.mail.from=john.doe
report.mail.password=123456789
report.mail.server=mail.company.com
report.mail.username=john
#Team Server Settings
tcm.server.accountLogin=true
tcm.server.enabled=true
tcm.server.name=ntp.company.com
tcm.server.password=123456789
tcm.server.port=18888
tcm.server.username=team_user
```

下記の値が定義されていない場合、DTP は自動的に値を設定します。Data Collector や Mail Server のポートなどのように、DTP の全般的な構成に基づいて値が設定されます。

```
#Report Center Settings
grs.data.port
#License Settings
license.network.host
license.network.port
#Mail Settings
report.mail.domain
report.mail.from
report.mail.password
report.mail.server
report.mail.username
#Team Server Settings
tcm.server.name
tcm.server.port
```

手動で値を設定すると、自動的に設定された値が上書きされます。詳細については『Parasoft Test ユーザーズ ガイド』を参照してください。





# ワークフロー、権限、承認プロセスの構成

Development Testing Platform (DTP) はプロセス コントロールを促進します。プロセス コントロールは法規制コンプライアンスを実現しようとする組織にとって特に重要です。DTP によって、開発チームは承認のワークフローとトラッキングを正確かつ簡単に設定してコントロールできます。

たとえば：

- どの成果物のステータス遷移（たとえばドラフトから承認、など）が電子署名あるいは変更理由の記録を必要とするか。
- ステータスの変更（たとえば進行中から修正済み、など）があったときに、レビュー タスクを生成すべきかどうか（そしてそのようなレビュー タスクを分配すべきかどうか）。
- プロジェクト成果物（要求、シナリオ、欠陥、テスト セッション）に対して、各ユーザーまたはユーザーの役割がどのようなステータス遷移を行うことができるか。たとえば、要求を作成したり欠陥をクローズする権限を一部のユーザーだけに制限します。
- 特定のステータス（たとえば検証済みや承認）を変更できるかどうか。

法規制コンプライアンスの要求を満たす必要があるプロセスにとって、これは特に重要です。

監査を容易にするために、すべての電子署名、変更理由、および成果物（要求、欠陥、タスクなど）に関連する変更がトラッキングされ、集約された場所からアクセスすることができます。

目的	参照先
プロジェクトまたはプロジェクト成果物が削除されないようにする。	321 ページの「制限モードでの権限とステータス遷移の設定」
各ユーザーに権限を設定して、ユーザーが実行できる成果物のステータス遷移をコントロールする（たとえば、特別なユーザーだけが " 検証済み " ステータスを設定できるようにする、など）。	321 ページの「制限モードでの権限とステータス遷移の設定」
電子署名の入力を必要とする、成果物のステータス遷移（たとえばドラフトから承認、など）を指定する。	280 ページの「電子署名と変更理由の設定」
変更理由の記録を必要とする、成果物のステータス遷移（たとえばドラフトから承認、など）を指定する。	280 ページの「電子署名と変更理由の設定」
ステータスの変更（たとえば進行中から修正済み、など）を指定するときに、イベント通知を行う。	282 ページの「レビュー タスクを設定する」
特定のステータス（たとえば検証済み、承認済み、など）が成果物に割り当てられたら、その成果物を変更できないようにする。	274 ページの「カスタム ステータスの追加」



目的	参照先
すべての電子署名、変更理由、および特定の成果物に関連する他の変更をトラッキングする。	242 ページの「[ 変更履歴 ] タブによる変更のトラッキング」

## [ 変更履歴 ] タブによる変更のトラッキング

すべての電子署名、変更理由、および成果物（要求、欠陥、タスクなど）に関連する他の変更を見るには、その成果物を開いて [ 変更履歴 ] タブに記録されたデータを参照します。

例：

タスクの編集: [36] 要件 27 の実装: 音声認識

全般 | **変更履歴** | コード 506 | 自動テスト 0 | シナリオ 0

変更履歴

作成者: admin  
作成日時: 2010-10-13 17:25:41

変更されたフィールド	削除された値	追加された値
変更時刻: [2010-10-14 13:32:33] by admin		
見送り残作業時間	9d	5d 3h 58m
変更時刻: [2010-10-13 17:26:59] by admin		
ステータス	オープン	作業中
反復 ID		8.0
実作業時間	0d 0h 0m	2d 6h 30m
監督者		
見送り残作業時間	0d 0h 0m	9d
変更時刻: [2010-10-13 17:25:41] by admin		
オーナー		kylie
ステータス		オープン
テストが必要		true
プロジェクト		PolicyCdemo
予定作業時間		9d
優先度		中
名前		要件 27 の実装: 音声認識
要求 ID		27
開始予定日		2010-10-11
関係		要件の実装

アクティビティ

1 アイテム

ユーザー名	開始日	終了日	継続時間
kylie	2010-10-13 17:26		519d 6h

# Report Center と Project Center のコンポーネント

このセクションでは、Report Center と Project Center のコンポーネントの基本事項について説明します。

基本的なサブシステムには次のものが含まれます。

- DTP Server: Tomcat Web サーバー エンジンを使用
- データベース エンジン (MySQL または Oracle)
- Data Collector
- JMS イベント ブローカー

サーバー サイドでこれらのコンポーネントが使用するデフォルトのポート番号は、次の表のとおりです。

コンポーネント	使用ポート
DTP Server	URL は、ポート 80 (Windows) またはポート 8080 (Linux) で実行する Tomcat レポート サーバーに接続するために使用されます。 <ul style="list-style-type: none"> <li>• http://[hostname-of-server]:80 (Windows)</li> <li>• http://[hostname-of-server]:8080 (Linux)</li> </ul>
MySQL database engine	3306 ( デフォルト )
Oracle database engine	1521 ( デフォルト )
Data Collector	32323
JMS イベント ブローカー	61617 ( デフォルト )

640 ページの「Data CollectorReport Center と Project Center のネットワークの検証」では、netstat コマンドを使って、プロセスがポート番号を使用しているか、つまりプロセスが実行しているかを検証する方法について説明しています。netstat コマンドは、Report Center を実行する前にポートの競合がないかを確認するのに役立ちます。

DTP と Data Collector は、起動されるとすぐにデータベース (MySQL または Oracle) に接続します。したがって、データベースが先に起動して最後にシャットダウンするよう、データベース エンジンが構成されます。理想的には、DTP と Data Collector だけが直接データベース エンジンを使用すべきです。ツールは、サーバーで Data Collector にポート 32323 で接続してデータを送信します。そしてユーザーはシステムとやり取りし、Web ブラウザーを使って Tomcat レポート サーバーに接続してデータを参照します。Tomcat レポート サーバーはポート 80 (Windows) または 8080 (Linux) で実行されます。DTP の URL は次のとおりです。

- Windows の場合 : http://[hostname-of-server]:80
- Linux の場合 : http://[hostname-of-server]:8080

まれなケースですが、必要に応じて DTP Web サーバー (287 ページの「Web サーバーのポート番号の変更」)、Data Collector (287 ページの「Data Collector のポート番号の変更」)、または JMS イベント ブローカー (599 ページの「JMS イベント ブローカー ポートの設定」) のデフォルトのポート番号を変更できます。

---

## データベース接続の変更

Development Testing Platform (DTP) では、異なるホストとポートでデータベースを使用することができます。データベースの設定は、Web UI から行うか (245 ページの「Web からのデータベース接続の変更」を参照)、DTP\_HOME\conf\ ディレクトリにある PSTRootConfig.xml ファイルを編集して行うことができます (246 ページの「PSTRootConfig.xml の編集によるデータベース接続の変更」を参照)。

---

## GRS および DTP データベースについて

DTP は 2 種類のデータベースにデータを格納します。GRS データベースと DTP データベースです。

---

### GRS データベース

GRS データベースは、DTP の前身である Concerto が収集したデータを格納します。Concerto から DTP にアップグレードしていて、まだ Concerto データにアクセスする必要がある場合、GRS データベースに必ず接続する必要があります。また、DTP ツールに関連する機能や今後の DTP のリリースで実装される機能を利用するには、DTP データベースへの接続を設定する必要があります。

---

### DTP データベース

GRS データベースと同じ種類のデータを保管するのに加えて、DTP データベースは Parasoft Static Analysis Execution Engine からのデータを扱います。Parasoft Static Analysis Execution Engine には、C/C++、Java、および .NET に対応した実行エンジンが含まれます。今後、DTP の機能は DTP データベースのデータを使用する予定です。

---

### データベースを共有してはいけない

データに関連する問題を回避するために、DTP Server をそれぞれ独自のデータベースに接続してください。

---

## Web からのデータベース接続の変更

1. Report Center で [ 管理 ] ドロップダウンメニューから [Report Center] を選択します。
2. [ 設定 ] > [ データベース ] を選択します。
3. 必要に応じて接続の設定を変更し、[ チェック ] ボタンをクリックしてデータが正確であることを検証します。
4. [ 保存 ] ボタンをクリックします。

Connected successfully to the GRS database. Connected successfully to the DTP database.

### Database Settings

**GRS Database Connection Settings**

Database type  MySQL  Oracle

Database url  
JDBC url definition mode:  Simple  Advanced

Host \*

Port \*

DB name \*

User and Password

User \*

Password

**DTP Database Connection Settings**

Database type  MySQL  Oracle

Database url  
JDBC url definition mode:  Simple  Advanced

Host \*

Port \*

DB name \*

User and Password

User \*

Password

## PSTRootConfig.xml の編集によるデータベース接続の変更

1. すべての Development Testing Platform サービスをシャットダウンします。
2. DTP\_HOME\conf\ ディレクトリに移動し、PSTRootConfig.xml を開きます。この xml ファイルには、には、DTP と Data Collector に共通のデータベース接続 パラメーターが定義されています。

### GRS データベース :

```
<db-type>MySQL</db-type>
```

```
<user>grs</user>
```

```

<password encrypted="true">abc123</password>
<!--
sample Oracle connection url: <url>jdbc:oracle:thin:@HOST:PORT:SID</url>
--->
<!--
sample MySQL connection url: <url>jdbc:mysql://HOST:PORT/DATABASE</url>
--->
<url>jdbc:mysql://localhost:3306/GRS</url>

```

**DTP データベース :**

```

<db-type>MySQL</db-type>
<user>dtp</user>
<password encrypted="true">abc123</password>
<!--
sample Oracle connection url: <url>jdbc:oracle:thin:@HOST:PORT:SID</url>
--->
<!--
sample MySQL connection url: <url>jdbc:mysql://HOST:PORT/DATABASE</url>
--->
<url>jdbc:mysql://localhost:3306/GRS</url>

```

**注意 :** Oracle と MySQL の URL の例がコメントで入っています。

3. DTP と Data Collector の両方のデータベース接続を再定義します。

以下は Oracle データベースの場合の例です。

```

<db-type>Oracle</db-type>
<user>grs</user>
<password encrypted="true">abc123</password>
<!--
sample Oracle connection url: <url>jdbc:oracle:thin:@HOST:PORT:SID</url>
--->
<!--
sample MySQL connection url: <url>jdbc:mysql://HOST:PORT/DATABASE</url>
--->
<url>jdbc:oracle:thin:@arcadia.company.com:1521:sid1</url>

```

4. 変更したファイルを保存します。
5. DTP と Data Collector のサービスを再起動します。

---

## データベース認証パスワードの手動更新

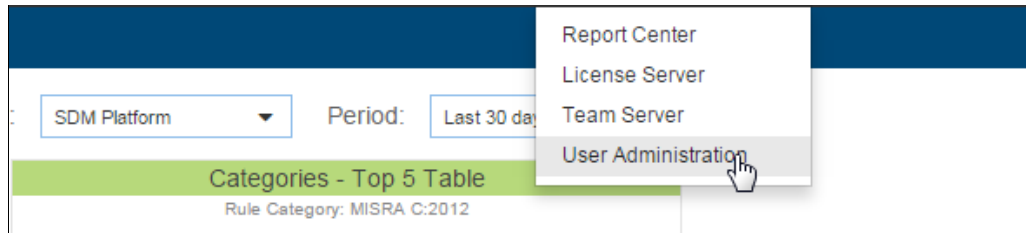
DTP UI で保存されるパスワードは、暗号化されて PSTRootConfig.xml に保管されます。データベース接続のために手動でパスワードを上書きする必要がある場合、password encrypted プロパティを false に変更して、プレーン テキストの値を設定することができます。

```
<password encrypted="false">plaintextpassword</password>
```

パスワードは、ブラウザー UI では隠され続けますが、暗号化はされません。セキュリティ上の理由から、暗号化されたパスワードが保管されるよう、データベース接続ページから認証設定を保存する必要があります。

## [User Administration] ページ

管理者権限があるユーザーは、[User Administration] ページにアクセスできます。[User Administration] ページでは DTP ユーザーに権限を許可したりユーザー権限を管理することができます。



DTP の管理を担当するユーザーは、セキュリティ設定の最初に管理権限を持つべきです。管理権限には以下が含まれます。

- Basic 権限 (pstsec\_basicAccess:true):  
この権限を定義して設定した場合、Security モジュールへのアクセスが許可されます。管理者は、定義されたユーザーと権限グループを編集できます。
- Administration 権限 (pstsec\_administration:true):  
この権限を定義して設定した場合、管理者は権限を編集して [ユーザー] セクションを編集できます。

データベースには、管理者権限を持つ "admin" というユーザーがすでに存在します。セキュリティ上の理由から、固有のパスワードを持った選ばれたユーザーに管理権限を割り当てることを推奨します。

### ユーザーの設定と管理に関する用語

以下の表は、User Administration モジュールの説明で使用する用語とその定義の一覧です。:

用語	定義
権限 (Permission)	<p>選択した DTP アプリケーションのユーザー権限を示します。権限は、tool:name:value の形式になっており、tool は権限を適用するツール名、name は権限タイプの名前、value は権限の値を意味します。</p> <p>たとえば、以下の権限は、ユーザーがプロジェクト Core に関する Report Center のデータにアクセスできることを意味します。</p> <pre>grs:project:Core</pre> <p>権限の値を正規表現で指定することによって、作業を簡略化できます。たとえば、プロジェクトの以前のバージョンや将来のバージョンのデータへのアクセス権を付与する必要があるとします。この場合、次のように指定できます。</p> <pre>grs:project:Core \d\\.d</pre> <p>すると、Core 1.0、Core 1.1 などへのアクセスが許可されます。</p> <p>権限は、権限グループおよびユーザーに適用されます。</p>



用語	定義
ネイティブ権限 (Native Permissions)	権限は権限グループに許可されます。
継承された権限 (Inherited Permissions)	親権限グループから継承された権限。
権限グループ (Permission Group)	権限のセット。権限グループは、複数のネイティブ権限 (Native Permission) を持つことができます。さらに、各権限グループは複数の親権限グループを持つことができます。  権限グループにおいて、Native Permission と Inherited Permission の両方を有効 / 無効に設定できます。有効 / 無効を設定することは、権限グループを継承してグループを作成して、継承された権限の中から特定の権限だけがが必要な場合に役立ちます。
ユーザー (User)	DTP ユーザー。権限グループに似ています。ユーザーは、複数の権限 (Native Permission) を持つことができ、複数の権限グループに属することができます。  ユーザーの Inherited Permission はグループ化されており、権限グループの階層を反映します。あらゆる権限をニーズに合わせて有効 / 無効に設定できます。異なる権限グループからユーザーによって継承された権限は分離されますが、個々の権限とリンクされます。
パースペクティブ (Perspective)	ユーザーが利用できる Report Center レポートと機能のセット。

DTP ユーザーの用語については、258 ページの「権限のシナリオ」を参照してください。

## 新規ユーザーの作成

新規ユーザーをデータベースに追加するには、ユーザーのアカウントを作成する必要があります。さらに、検索、編集、削除といったアクションについてユーザーの権限を許可して管理しなければなりません。

1. [ユーザー] > [新規追加] を選択します。
2. 新規ユーザーの情報を入力します (ログインとパスワードは必須です)。

デフォルトでは、ユーザーのログイン名は大文字と小文字を区別します (たとえば、JohnSmith、johnSmith、johnsmith はすべて別のユーザーです)。\$DTP\_HOME/conf/PSTRootConfig.xml ファイルで以下の変更を行って DTP Server サーバーを再起動することで、ユーザー ログイン名の大文字 / 小文字を区別しないよう設定を変更できます。

```
<root-config>/<case-sensitive>>true</case-sensitive>
```

を以下のように変更します。

```
<root-config>/<case-sensitive>>false</case-sensitive>
```

この設定は、ユーザー名に関連するすべての機能に影響します（たとえば、ログイン処理、Parasoft Test の自動テストの結果、Source Scanner など）。

3. [保存] をクリックしてユーザーを作成します。[権限とグループ] タブが表示されます。

User new\_user has been created successfully.

---

General Permissions & Groups

Login new\_user

Password .....

Name New

Surname User

Email new@user.com

Save Reset Cancel Delete

---

group preview

General	Permissions	Hierarchy	Members
Name	Native	There are no ancestors	There are no members
Description	Inherited		
Type User-defined group	There are no permissions		

4. [権限とグループ] タブをクリックします。
5. [グループメンバーシップ] セクションで、すべて展開ボタン (🔍) をクリックし、新しいユーザーの権限を確認します。

**Group Membership**

🔍 PST Basic Access

Tool	Name	Value	Regexp
<input checked="" type="checkbox"/> grs	basicAccess	true	
<input checked="" type="checkbox"/> ls	basicAccess	true	
<input checked="" type="checkbox"/> pst	basicAccess	true	
<input checked="" type="checkbox"/> pstsec	basicAccess	true	
<input checked="" type="checkbox"/> reports	basicAccess	true	
<input checked="" type="checkbox"/> tcm	basicAccess	true	

デフォルトではユーザーには Basic Access が割り当てられ、ユーザーは DTP にログインすることができます。組織でのユーザーの役割に合った権限をさらに許可することができます。詳細については 253 ページの「ネイティブ権限の割り当て」を参照してください。

6. 権限を割り当てて [保存] をクリックします。

## プロジェクトへのユーザーの追加

ユーザーを作成して権限とグループを割り当てても、自動的にユーザーがプロジェクトに追加されるわけではありません。ユーザーをプロジェクト チームに追加し、Report Center のプロジェクト管理でプロジェクト チームをプロジェクトに追加する必要があります。チームへのユーザーの追加とプロジェクトへのチームの追加に関するワークフロー特有の手順については、230 ページの「プロジェクトの作成と構成」を参照してください。

## ユーザーの編集

1. [ユーザー] > [検索] を選択します。
2. 検索結果の表で、編集するユーザーの [編集] ボタンをクリックします。

Login	Name	Surname	Email	
Haryu	Harry Yu	Yu	haryu@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
JSupport Mail Box	Jtest Support Mail Box	Support Mail Box	root@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
JtestSupport	JTest Support	Support	sysadmin@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
MLunch Room	Monrovia Lunch Room	Lunch Room		<a href="#">Edit</a> <a href="#">Delete</a>
adhikari	Bhavya Adhikari	Adhikari	adhikari@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
adurbha	Anama Durbhakula	Durbhakula	adurbha@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>

3. [全般] > [権限とグループ] タブをクリックし、変更を行います。DTP の機能へのアクセスを変更する方法については、253 ページの「ネイティブ権限の割り当て」を参照してください。
4. [保存] をクリックして変更を保存します。

## ユーザーの削除

1. [ユーザー] > [検索] を選択します。
2. 検索結果の表で、削除するユーザーの [削除] ボタンをクリックします。

Login	Name	Surname	Email	
Haryu	Harry Yu	Yu	haryu@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
JSupport Mail Box	Jtest Support Mail Box	Support Mail Box	root@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
JtestSupport	JTest Support	Support	sysadmin@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
MLunch Room	Monrovia Lunch Room	Lunch Room		<a href="#">Edit</a> <a href="#">Delete</a>
adhikari	Bhavya Adhikari	Adhikari	adhikari@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>
adurbha	Anama Durbhakula	Durbhakula	adurbha@parasoft.com	<a href="#">Edit</a> <a href="#">Delete</a>

3. ユーザーが完全に削除されることについて確認を求められたら、削除を承認します。

## 権限グループへのユーザーの割り当て

権限の範囲が定義された権限グループにユーザーを割り当てることができます。この方法は、プロジェクトグループに権限を許可する場合に便利です。DTP にはビルトイン権限グループがあらかじめ用意されていますが、組織に合った権限グループの作成が必要な場合もあります。グループの作成方法については 255 ページの「権限グループの作成」を参照してください。

1. ユーザーのプロファイル ページを開きます。252 ページの「ユーザーの編集」を参照してください。
2. [ 権限とグループ ] タブをクリックします。
3. [ グループの追加 ] セクションで [ 検索 ] をクリックして、すべてのアクセス可能なグループを表示します。

The screenshot shows the 'Permissions & Groups' interface. On the left, there's a 'missions' section with 'Add New Permission' options for Tool, Name, and Value. The main area is 'Group Membership', which includes a search box and a list of groups. The 'Add Group' section is also visible, showing a search button and a list of groups with expand/collapse icons.

4. 追加ボタン ( ) をクリックしてグループにユーザーを追加します。また、[ グループ メンバーシップ ] セクションで削除ボタン ( ) をクリックして、グループからユーザーを削除することもできます。自動的に同期化されるように設定されたグループにユーザーを追加することはできません。詳細については 754 ページの「ユーザー情報に基づいたグループ メンバーシップの同期化」を参照してください。
5. [ 保存 ] をクリックします。

## ネイティブ権限の割り当て

権限グループにユーザーを割り当てるのに加えて (252 ページの「権限グループへのユーザーの割り当て」)、個人ベースでユーザーに DTP の機能を許可することもできます。

1. ユーザーのプロファイル ページを開きます。252 ページの「ユーザーの編集」を参照してください。

2. [ 権限とグループ ] タブをクリックします。

General | **Permissions & Groups**

**Native Permissions**

*There is no permissions.* **Add New Permission**

Tool <Select Tool> ▾

Name *(Tool is not selected)*

Value *(Tool and Name are not selected)*

Add

**Groups Membership**

**Add Group**

Name

Search

*There is no search results*

Save | Reset | Cancel | Delete

3. [ ツール ] ドロップダウン リストから DTP の機能を選択します。
4. [ 名前 ] ドロップダウン リストから、許可する権限の範囲を表す名前を選択します。
5. [ 値 ] から true または false を選択してツールへのアクセスを許可または拒否します。
6. [ 追加 ] をクリックして設定を完了します。また、削除ボタン (✖) をクリックしてネイティブ権限を削除することもできます。

malgosia

Permission grs:reportCategory:Architect Dashboard has been added successfully.

**User is not saved!**

General | **Permissions & Groups**

**Native Permissions**

Tool	Name	Value
✖ grs	reportCategory	Architect Dashboard

**Add New Permission**

Tool grs ▾

Name reportCategory ▾

Value Architect Dashboard ▾

Add

**Groups Membership**

**Add Group**

Name

Search

*There is no search results*

Save | Reset | Cancel | Delete

7. [ 保存 ] をクリックしてネイティブ権限を追加します。

## 自動的な同期化

"Use Member Attribute" オプションが有効な場合に、手動で作成したグループを、相当する LDAP グループと自動的に同期化するように設定することができます (754 ページの「メンバーシップ検索設定」を参照)。自動的な同期化の際、DTP は以下の処理を行います。

- 相当する LDAP グループのメンバーに基づいて、このグループに割り当てられたユーザーのリストを定期的に更新します。
- ユーザーがまだ存在しない場合、ユーザーを自動的に作成します。
- グループが新しい場合、あるいは自動的な同期化が無効だった場合に、変更を保存した後、直ちにグループを同期化します。

複数の LDAP サーバーを使用するように DTP が設定されている場合、グループの同期化は最初に一致した LDAP グループに基づきます。

## 権限グループについて

プロジェクト チームでのユーザーの役割に基づいて特定の権限を割り当てるには、新しい権限グループを作成します。権限グループには一連の権限が含まれ、その権限によってユーザーは特定のアクションを実行したり特定のレポートとプロジェクト データにアクセスしたりすることができます。

一部のビルトイン グループ権限には、特定の役割が含まれています。ユーザーは、プロジェクト関連のグループに割り当てる前に、まずデフォルト グループに割り当てなければなりません。デフォルト グループは、設定された権限の範囲によって、ユーザーの役割を表します。

## 権限グループの作成

グループの階層と新規グループの作成の詳細については、263 ページの「権限グループ階層の調整」を参照してください。

1. [グループ] > [新規追加] を選択します。
2. グループの名前と情報をそれぞれ適切なフィールドに入力します。

The screenshot shows the 'New Group' form in the PARASOFT Development Testing Platform User Administration interface. The form is titled 'New Group' and has a blue header bar with the text 'PARASOFT Development Testing Platform User Administration'. Below the header bar are three tabs: 'Users', 'Groups', and 'Directories'. The 'Groups' tab is selected. The form contains two input fields: 'Name' with the value 'New Group' and a red asterisk indicating it is required, and 'Description' with the value 'Details about New Group'. At the bottom of the form are three buttons: 'Save', 'Reset', and 'Cancel'.

グループを同期化できるように DTP を設定している場合、[Synchronize automatically] オプションを使用できます。グループの作成後に直ちにグループを同期化するには、グループの作成時にこのオプションを有効にします。詳細については 754 ページの「メンバーシップ検索設定」を参照してください。

3. [保存] をクリックしてグループを作成し、[権限と階層] および [Users] タブを有効にします。
4. [権限と階層] タブをクリックします。
5. グループに権限を追加し (256 ページの「権限グループの設定」)、階層を設定します (257 ページの「権限グループの編成」)。
6. [保存] をクリックしてグループの設定を完了します。

## 権限グループの編集

1. メインメニューから [グループ] > [検索] を選択します。
2. 検索結果の表で [編集] をクリックして詳細ページを開きます。

### Search Groups

Name	Description	
New Group	Details about New Group	<a href="#">Edit</a> <a href="#">Delete</a>
PST Administration	Group defines admin permissions for PST	
PST Basic Access	Group defines basic permissions for PST	
Parasoft Contact Group	Parasoft Contact Group	<a href="#">Edit</a> <a href="#">Delete</a>
Sample Strict Mode Permissions		<a href="#">Edit</a> <a href="#">Delete</a>
Sample Team		<a href="#">Edit</a> <a href="#">Delete</a>

3. グループに対して変更を加えます。256 ページの「権限グループの設定」および 257 ページの「権限グループの編成」を参照してください。
4. [保存] をクリックします。

## 権限グループの設定

1. 権限グループの詳細ページを開きます。256 ページの「権限グループの編成」を参照してください。
2. [権限と階層] タブをクリックします。
3. [ツール] ドロップダウン リストから DTP の機能を選択します。
4. [名前] ドロップダウン リストから、許可する権限の範囲を表す名前を選択します。
5. [値] から true または false を選択してツールへのアクセスを許可または拒否します。

6. [追加] をクリックして、アクセスする機能を追加します。また、削除ボタン (✖) をクリックして機能へのアクセスを削除することもできます。

**New Group**  
 Permission pstsec:administration:true has been added successfully.  
 Group is not saved!

General | **Permissions & Hierarchy** | Users

**Permissions**

**Native**

Tool	Name	Value	Regexp
✖ pstsec	administration	true	
✖ reports	administration	true	

**Add New Permission**

Tool: pstsec  
 Name: administration  
 Value: true  
 Add

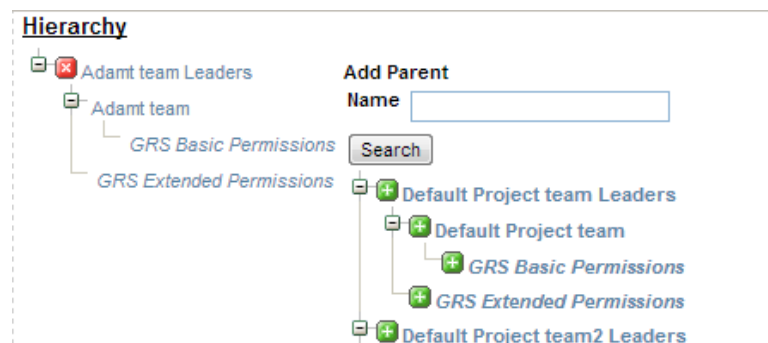
**Inherited**  
 There are no permissions

7. 必要に応じてステップ 5 - 8 を繰り返し、[保存] をクリックして権限グループの設定を完了します。

## 権限グループの編成

グループは階層的に編成できます。階層化することで、「変更されない権限を持ったグループ」と「よく変更される権限を持ったグループ」を分けることができます。変更されない権限は、たとえば Report Center レポートの可用性を示す権限です。よく変更される権限は、たとえば Report Center プロジェクトの可用性を示す権限です。グループは、個々に権限を持つ場合もあれば、先祖から権限を継承する場合もあります。

1. 権限グループの詳細ページを開きます。256 ページの「権限グループの編集」を参照してください。
2. [権限と階層] タブをクリックします。
3. デフォルトでは、新規グループは親ノードとして追加されます。グループ ツリーで追加ボタン (+) をクリックして、ノードを親としてグループに追加します。また、削除ボタン (✖) をクリックして親ノードを削除することもできます。





## 4. 親グループから継承する権限を選択します。

New Group

Group Adamt team Leaders was successfully added as the parent of Group New Group.

Group is not saved!

General Permissions & Hierarchy Users

**Permissions**

Native

Tool	Name	Value	Regexp
<input checked="" type="checkbox"/>	pstsec	administration	true
<input checked="" type="checkbox"/>	reports	administration	true

Add New Permission

Tool: pstsec

Name: administration

Value: true

Add

**Inherited**

Tool	Name	Value	Regexp
<input checked="" type="checkbox"/>	grs	perspective	Basic
<input checked="" type="checkbox"/>	grs	perspective	Efficiency
<input checked="" type="checkbox"/>	grs	perspective	Management
<input checked="" type="checkbox"/>	grs	perspective	QA
<input checked="" type="checkbox"/>	grs	perspective	Technical
<input checked="" type="checkbox"/>	grs	perspective	Trends
<input checked="" type="checkbox"/>	grs	project	Adamt
<input checked="" type="checkbox"/>	grs	project	Adamt2

**Hierarchy**

Adamt team Leaders Add Parent

Name:

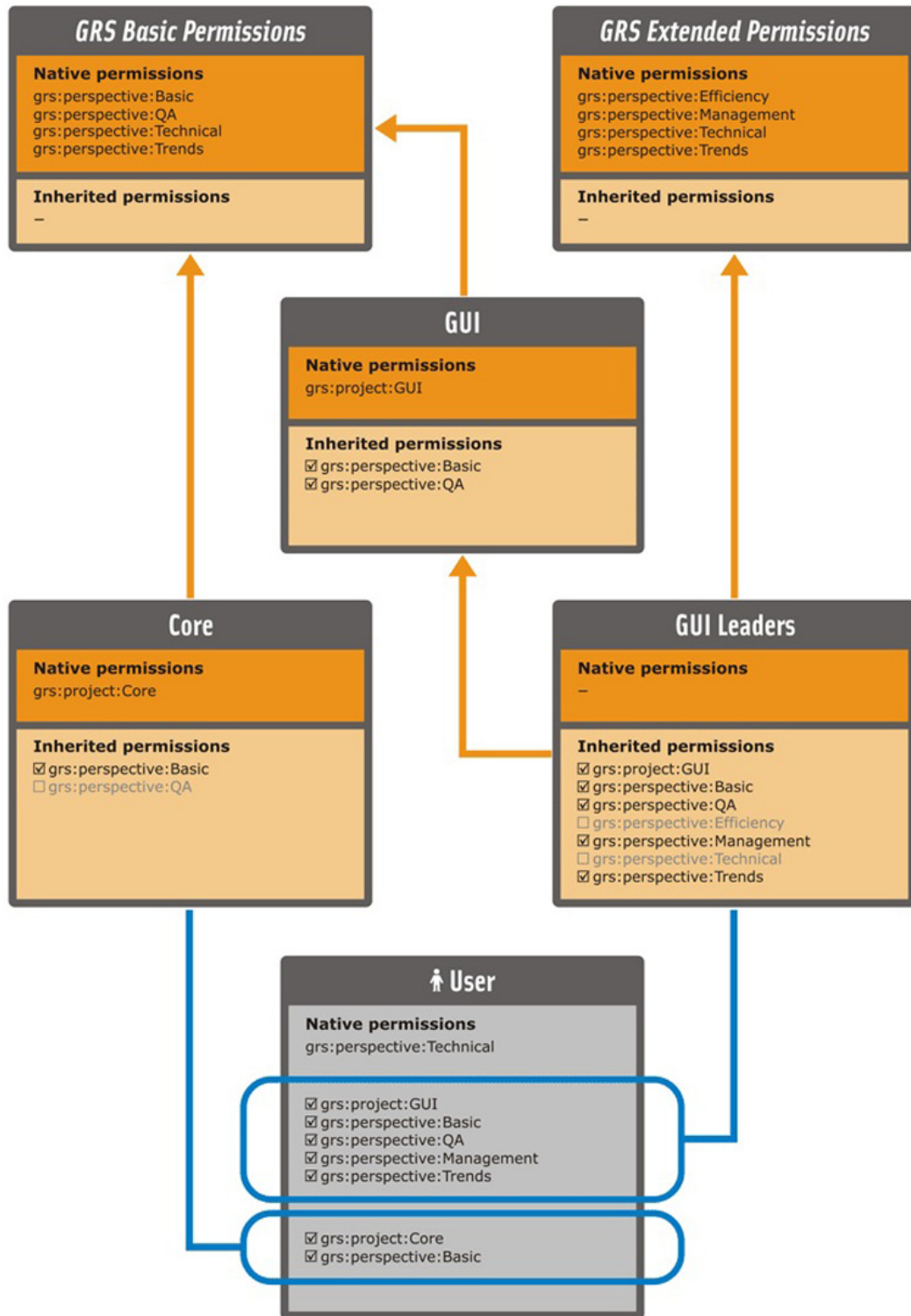
Search

- Adamt team Leaders
  - Adamt team
    - GRS Basic Permissions
    - GRS Extended Permissions
  - Default Project team Leaders
    - Default Project team
      - GRS Basic Permissions
      - GRS Extended Permissions
    - Default Project team2 Leaders
      - Default Project team2
        - GRS Basic Permissions
        - GRS Extended Permissions

## 5. [保存] をクリックして変更を保存します。

## 権限のシナリオ

以下の図は、権限付与のシナリオを図で表したものです。



---

## シナリオの背景

以下の条件が Report Center で当てはまるものとします。

- ユーザー "jdoe" はプロジェクト Core の開発者です。
- ユーザー "jdoe" は開発者ダッシュボード、単体テストおよび静的解析レポートにアクセスする必要があります。
- ユーザー "jdoe" は GUI プロジェクトのリーダーでもあるので、アーキテクチャおよびマネージャー ダッシュボードにもアクセスできる必要があります。

プロジェクト項目 (データ、レポート、ダッシュボード) へのアクセスは完全に Report Center の権限に依存しています (267 ページの「権限とパースペクティブ」を参照)。DTP の管理者が行うべきタスクは、ユーザー "jdoe" に適切なアクセス権を与えて Core および GUI プロジェクトのデータにアクセスできるようにすることです。

上記の例に基づいて検討すると、"jdoe" は QA、マネージメント、およびトレンド パースペクティブをととして利用できるプロジェクト データを必要としています。"jdoe" に必要なアクセス権を与えるために、このユーザーのために特別に設計された権限グループを作成できます。その際、ビルトイン グループ (265 ページの「ビルトイン グループ」を参照) や他の新たに作成したグループを利用できます。ビルトイングループである GRS Basic Permission グループと、プロジェクト権限が割り当てられた新しいグループである Core および GUI グループを利用します。

Core グループを作成したら、grs:perspective:QA 権限を無効にします。なぜなら、"jdoe" は テスター ダッシュボードや手動テストにアクセスする必要はないからです。次に、GUI グループ (新規) および GRS Extended Permission group (ビルトイン) を継承する GUI Leaders グループを作成します。

その後、リーダーにとって重要ではない grs:perspective:Efficiency および grs:perspective:Technical 権限を無効にします。

グループを作成したら、"jdoe" を Core および GUI Leaders グループのメンバーとして追加します。グループのメンバーとなることで、"jdoe" には 2 つの異なる権限グループが与えられます。つまり、Jdoe がマネージャー ダッシュボードを参照しているときには、grs:project:GUI および grs:perspective:Management 権限があるので GUI プロジェクトのレポートを参照できますが、Core プロジェクトのレポートは参照できません。

"jdoe" に個別にテクニカル権限 (grs:perspective:Technical) を与えると、どちらの権限グループにも適用されます。つまり、"jdoe" は Core プロジェクトと GUI プロジェクトのテクニカル レポートを参照できます。

---

## ユーザーと権限グループの追加

以下の例では、ユーザー "jdoe" が追加され、必要な権限が設定されます。

1. User Administration のメイン ページで、[ ユーザー ] > [ 新規追加 ] をクリックします。
2. ユーザーの情報を入力します (ログイン名の大文字 / 小文字は区別されます)。[ 保存 ] をクリックします。
3. [ グループ ] > [ 新規追加 ] をクリックします。同期化オプションが有効な場合、グループからユーザーを追加したり削除したりすることはできません (255 ページの「自動的な同期化」を参照)。

4. "Core" というグループを作成し、[ 保存 ] をクリックします。このグループには、プロジェクトのレポートとデータへのアクセスを "jdoe" に許可するために必要なすべての権限が含まれます。

**New Group**

**Name** Core \*

**Description** Group for Core project

Save Reset Cancel

5. [ 権限と階層 ] タブをクリックし、親として GRS Basic Permissions グループを追加します。次の操作を行います。

- a. [ 親の追加 ] の [ 名前 ] フィールドに「GRS Basic Permissions」と入力して [ 検索 ] ボタンをクリックします。

GRS Basic Permissions ノードが表示されます。

Core Logout

General **Permissions & Hierarchy** Users

**Permissions**

**Native**  
There are no permissions

**Add New Permission**

Tool <Select Tool>

Name (Tool is not selected)

Value (Tool and Name are not selected)

Add

**Inherited**

	Tool	Name	Value	Regex
<input checked="" type="checkbox"/>	grs	perspective	Basic	
<input type="checkbox"/>	grs	perspective	QA	

Save Reset Cancel Duplicate Delete

**Hierarchy**

↳ GRS Basic Permissions **Add Parent**

Name GRS Basic Permissions

Search

↳ Core

↳ GRS Basic Permissions

**GRS Basic Permissions group preview**

**General**

Name GRS Basic Permissions

Description Group defines basic permissions for GRS

Type Built in group

**Permissions**

**Native**

Tool	Name	Value	Regex
grs	perspective	Basic	
grs	perspective	QA	

**Inherited**

There are no permissions

**Hierarchy**

There are no ancestors

**Members**

There are no members

- b. GRS Basic Permissions ノードの緑色の [ + ] アイコンをクリックして、グループに対して新しい階層を参照します。

- n. [継承] セクションで [grs:perspective:QA] のチェックボックスをオフにして [保存] ボタンをクリックします。

**Permissions**

**Native**  
There are no permissions

**Add New Permission**

Tool: grs  
Name: perspective  
Value:   regexp  
Predefined values: Basic

Add

**Inherited**

	Tool	Name	Value	Regex
<input checked="" type="checkbox"/>	grs	perspective	Basic	
<input checked="" type="checkbox"/>	grs	perspective	QA	
<input checked="" type="checkbox"/>	grs	perspective	Technical	
<input checked="" type="checkbox"/>	grs	perspective	Trends	

6. Core グループにネイティブ権限を追加します。
- [権限と階層] タブで、[新規権限ツールの追加] ドロップダウン リストから [grs] を選択します。
  - [新規権限名の追加] ドロップダウン リストから [project] を選択します。
  - [Predefined values] ドロップダウン メニューから Core (プロジェクト名) を選択します。Core が表示されていない場合、[値] フィールドで「Core」と入力することができます (プロジェクトの作成の詳細については 582 ページの「ラン コンフィギュレーション新規プロジェクトの追加」を参照)。
  - [追加] ボタンをクリックしてから [保存] ボタンをクリックします。
7. Core グループのメンバーとしてユーザー jdoe を追加します。
- [ユーザー] タブをクリックし、[ログイン名] フィールドに「jdoe」と入力します。

**Core**

General Permissions & Hierarchy **Users**

**Members**

	Login	Name	Surname	Email
→ x	jdoe	John	Doe	jdoe@company.com

**Add**

Login: jdoe

Search

	Login	Name	Surname	Email
→ +	jdoe	John	Doe	jdoe@company.com

Save Reset Cancel Duplicate Delete

**jdoe user preview**

**General**  
Login: jdoe  
Name: John  
Surname: Doe  
Email: jdoe@company.com

**Native Permissions**  
There are no permissions

**Groups Membership**  
Core  
PST Basic Access

- o. [ 検索 ] ボタンをクリックします。ユーザー jdoe の詳細情報が [ 検索 ] ボタンの下に表示されます。
- p. jdoe の緑色の [ + ] アイコンをクリックして、jdoe をメンバーとしてグループに追加し、[ 保存 ] をクリックします。

## 権限グループ階層の調整

ユーザー "jdoe" は GUI プロジェクトのリーダーでもあるため、適切な権限が必要です。

1. "GUI" というグループについて、260 ページの「ユーザーと権限グループの追加」のステップ 1-4 を行います。

The screenshot shows a form with the following fields and buttons:

- Name:** GUI \*
- Description:** Group of GUI project
- Buttons:** Save, Reset, Cancel

2. [ 権限と階層 ] タブをクリックし、GUI グループの親として GRS Basic Permissions グループを追加します。
  - a. [ 親の追加 ] の [ 名前 ] フィールドに「GRS Basic Permissions」と入力して [ 検索 ] ボタンをクリックします。GRS Basic Permissions ノードが返されます。

The screenshot shows the 'Permissions & Hierarchy' tab with the following sections:

- General** | **Permissions & Hierarchy** | **Users**
- Permissions**
  - Native:** There are no permissions
  - Add New Permission:**
    - Tool: <Select Tool>
    - Name: (Tool is not selected)
    - Value: (Tool and Name are not selected)
    - Buttons: Add
- Inherited:**

	Tool	Name	Value	Regexp
<input checked="" type="checkbox"/>	grs	perspective	Basic	
<input checked="" type="checkbox"/>	grs	perspective	QA	
- Hierarchy:**
  - [-] GRS Basic Permissions
- Buttons:** Save, Reset, Cancel, Duplicate, Delete

- b. GRS Basic Permissions ノードの緑色の [ + ] アイコンをクリックして、GUI グループに新しい権限の階層が許可されたことを確認し、[ 保存 ] をクリックします。
3. ステップ 2 を繰り返して、GUI Leaders 権限グループを追加します。このグループは GUI グループ (ステップ 2 で作成) と GRS Extended Permissions (ビルトイン グループ) を継承します。

4. 次の図のように、GUI Leaders グループに継承された権限のうち grs:perspective:Efficiency および grs:perspective:Technical のチェックボックスをオフにします。

**Permissions & Hierarchy**

**Native**

Tool	Name	Value	Regexp
<input checked="" type="checkbox"/>	grs	project	GUI

**Add New Permission**

Tool:

Name:

Value:   regexp

Predefined values:

**Inherited**

Tool	Name	Value	Regexp
<input type="checkbox"/>	grs	perspective	Efficiency
<input checked="" type="checkbox"/>	grs	perspective	Management
<input type="checkbox"/>	grs	perspective	Technical
<input checked="" type="checkbox"/>	grs	perspective	Trends

**Hierarchy**

- GRS Extended Permissions

5. GUI Leaders グループのメンバーとしてユーザー jdoe を追加します。
- User Administration ページで [ユーザー] > [検索] をクリックします。
  - [ログイン名] フィールドに「jdoe」と入力し、[検索] をクリックします。[検索] ボタンの下にユーザー jdoe の詳細情報が表示されます。
  - jdoe の緑色の [+] アイコンをクリックして jdoe をメンバーとしてグループに追加し、[保存] をクリックします。
6. jdoe の名前をクリックして詳細を表示し、jdoe に与えられた権限を確認します。

**Users**

**Members**

Login	Name	Surname	Email	
<input checked="" type="checkbox"/>	jdoe	John	Doe	jdoe@company.com

**Add**

Login:

Login	Name	Surname	Email	
<input checked="" type="checkbox"/>	jdoe	John	Doe	jdoe@company.com

**jdoe user preview**

**General**

Login: jdoe  
Name: John  
Surname: Doe  
Email: jdoe@company.com

**Native Permissions**

There are no permissions

**Groups Membership**

- Core
- GUI Leaders
- PST Basic Access

7. jdoe に ネイティブの Technical 権限 (grs:perspective:Technical) を与えます。この権限は Core グループと GUI グループの両方に適用されます。jdoe は 2 つのグループに属するプロジェクトのテクニカル レポートを参照できるようになります。

- 左側の [ユーザー] > [検索] をクリックし、ユーザー「jdoe」を検索します。
- ユーザー "jdoe" を編集します。
- [権限とグループ] タブをクリックし、[新規権限の追加] セクションを次の図のように編集します。

- [保存] をクリックします。

8. 再びユーザー「jdoe」を検索して最終的な権限を確認します。

Login	Name	Surname	Email
admin			
grs			
→ jdoe	John	Doe	jdoe@company.com
jdoe			
ziomal			

[Edit](#) [Delete](#)

#### jdoe user preview

##### General

**Login** jdoe  
**Name** John  
**Surname** Doe  
**Email** jdoe@company.com

##### Native Permissions

Tool	Name	Value	Regexp
grs	perspective	Technical	

##### Groups Membership

Core  
 GUI Leaders  
 PST Basic Access

## ビルトイン グループ

ユーザーとグループを簡単に設定できるように、DTP には一般的な権限があらかじめ設定されたビルトイン グループが用意されています。新規にグループを作成するときは、ビルトイン グループを親として使用することを推奨します。

**注意：**ビルトイン グループを編集することはできません。



各ビルトイン グループについて以下で説明します。

- **PST Basic Access:** 基本的な権限を定義します。新規作成されたユーザーは自動的にこのグループのメンバーになります。

このグループに属するユーザーは、Report Center と Project Center にログインできません。ただし、これらのモジュールの管理用コントロールにはアクセスできません。アクションを実行するには、さらに権限が必要です。

#### PST Basic Access group preview

##### General

**Name** PST Basic Access

**Description** Group defines basic permissions for PST

**Type** Built in group

##### Permissions

###### Native

Tool	Name	Value
grs	basicAccess	true
ls	basicAccess	true
pst	basicAccess	true
pstsec	basicAccess	true
tcm	basicAccess	true

- **PST Administration:** 管理権限を定義します。このグループのメンバーには、DTP 内の各アプリケーション (Report Center、Team Server、License Server、USERS) の管理権限が許可されます。

このグループに属するユーザーは、各 DTP アプリケーションの管理ページからデータを管理できます。

#### PST Administration group preview

##### General

**Name** PST Administration

**Description** Group defines admin permissions for PST

**Type** Built in group

##### Permissions

###### Native

Tool	Name	Value
grs	administration	true
ls	administration	true
pst	administration	true
pstsec	administration	true
tcm	administration	true

###### Inherited

	Tool	Name	Value
<input checked="" type="checkbox"/>	grs	basicAccess	true
<input checked="" type="checkbox"/>	ls	basicAccess	true
<input checked="" type="checkbox"/>	pst	basicAccess	true
<input checked="" type="checkbox"/>	pstsec	basicAccess	true
<input checked="" type="checkbox"/>	tcm	basicAccess	true

- **GRS Basic Permissions:** Report Center の基本的な権限を定義します。

このグループに属するユーザーは、Basic および QA パースペクティブ (270 ページの「パースペクティブの詳細」を参照) に基づいてユーザーに割り当てられたプロジェクトの

特定の Report Center レポートを参照できます。つまり、ユーザーがプロジェクトに割り当てられていない場合は、どのレポートも参照できません。

#### GRS Basic Permissions group preview

##### General

**Name** GRS Basic Permissions

**Description** Group defines basic permissions for GRS

**Type** Built in group

##### Permissions

###### Native

Tool	Name	Value
grs	perspective	Basic
grs	perspective	QA

- **GRS Extended Permissions:** Report Center の拡張権限を定義します。

このグループに属するユーザーは、Efficiency、Management、Technical、および Trends パースペクティブ (270 ページの「パースペクティブの詳細」を参照) に基づいてユーザーに割り当てられたプロジェクトの特定の Report Center レポートを参照できます。つまり、ユーザーがプロジェクトに割り当てられていない場合は、どのレポートも参照できません。

#### GRS Extended Permissions group preview

##### General

**Name** GRS Extended Permissions

**Description** Group defines extended permissions for GRS

**Type** Built in group

##### Permissions

###### Native

Tool	Name	Value
grs	perspective	Efficiency
grs	perspective	Management

- **GRS Administrators:** Report Center の管理権限を定義します。

このグループに属するユーザーは、管理ページにアクセスして編集や管理を行うことができます。

#### GRS Administrators group preview

##### General

**Name** GRS Administrators

**Description** Group of GRS administrators.

**Type** Built in group

##### Permissions

###### Native

Tool	Name	Value
grs	administration	true

###### Inherited

## 権限とパースペクティブ

このセクションでは以下のユーザーとグループの情報について説明します。

- Development Testing Platform 権限
- パースペクティブの詳細

## Development Testing Platform 権限

権限	説明
pst:basicAccess:true	DTP にログインするにはこの権限が必要です。ただし、ユーザーがアクセスできる DTP 機能を指定するために、さらに権限が必要です。
pst:administration:true	ユーザーが DTP アプリケーションのデプロイと管理ができるよう、DTP Control Center へのアクセスを許可します。
pstsec:basicAccess:true	DTP Security アプリケーション ( Users Administration コンポーネント ) にログインするには、この権限が必要です。ただし、自分の個人データを変更することはできますが、他人の個人データを変更することはできません。
pstsec:administration:true	ユーザーおよび権限グループのデータの編集と変更を許可します。
grs:basicAccess:true	Report Center にログインするにはこの権限が必要です。ただし、ユーザーがアクセスできる Report Center を指定するには、さらに権限が必要です。
grs:administration:true	Report Center の管理ページへのアクセスを許可します。
grs:project:<project name or regular expression pattern>	Report Center に定義された特定のプロジェクトのデータへのアクセスを許可します。 例： grs:project:Core 権限は Core プロジェクトのデータへのアクセスを許可します。grs:project:Core \d\\.d という正規表現による権限は Core 1.0 および Core 1.1 プロジェクトへのアクセスを許可します。
grs:prioritizeAll:<project name>	プロジェクトに関連する違反の優先度を設定する権利を許可します。 チームのデフォルト権限：リーダー ( リーダーはメンバーから権限を継承します )
grs:prioritizeOwner<project name>	プロジェクトに関連する違反の優先度を設定する権利を、違反の所有者にのみ許可します。 チームのデフォルト権限：メンバー
grs:testSessionStatusChange: <status value to status value>	制限モードのみ (321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。 テストセッションに対して作業するときに、どのようなステータス遷移ができるかを指定します。

権限	説明
grs:reqStatusChange: <status value to status value>	制限モードのみ (321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。 要求に対して作業するときに、どのようなステータス遷移ができるかを指定します。
grs:defectStatusChange: <status value to status value>	制限モードのみ (321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。 欠陥に対して作業するときに、どのようなステータス遷移ができるかを指定します。
grs:testStatusChange: <status value to status value>	制限モードのみ (321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。 テストに対して作業するときに、どのようなステータス遷移ができるかを指定します。
grs:scenarioDeleteRestore: true	制限モードのみ (321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。 テスト シナリオをユーザーが削除および復元できることを指定します。
grs:scenarioStatusChange: <status value to status value>	制限モードのみ (321 ページの「制限モードでの権限とステータス遷移の設定」を参照)。 テスト シナリオに対して作業するときに、どのようなステータス遷移ができるかを指定します。
grs:viewSourceCode:<project name>	プロジェクトに関連するソース コードを参照する権利を許可します。 チームのデフォルト権限：メンバー
ls:basicAccess:true	今後の使用のため。ただし、この権限は DTP Basic Access 権限グループと共に許可されるはずです。 アプリケーションにアクセスする間、この権限は検証されません。
ls:administration:true	ライセンスの管理 (追加、削除、予約 など) を行うために License Server の管理ページにアクセスすることを許可します。
tcm:basicAccess:true	今後の使用のため。ただし、この権限は DTP Basic Access 権限グループと共に許可されるはずです。 アプリケーションにアクセスする間、この権限は検証されません。
tcm:administration:true	Team Server データを管理するための、Team Server の管理ページへのアクセスを許可します。たとえば、Team Server のデータ、作成したサンドボックス、テスト コンフィギュレーションのロードなどについてアクセスの許可と制限を行うことができます。

権限	説明
em:role:administration	Environment Manager のすべてのアクティビティへのアクセスを許可します。すなわち、権限のテスト、環境のセットアップ、システムと環境の定義、アクセス権の制御、およびテストデータの管理です。詳細については『Environment Manager ユーザーズ ガイド』を参照してください。
em:role:system	Environment Manager で、環境のセットアップおよびテストジョブの作成と実行を許可します。どちらのアクションもリソースへの適切な権限が必要です。em:role:system は、テスト データに対するすべてのリポジトリ アクションの実行も許可します。詳細については『Environment Manager ユーザーズ ガイド』を参照してください。
em:role:provision	Environment Manager で、ユーザーがアクセスできるソースに対して環境をセットアップする権限を許可します。em:role:provision は、テスト データへの読み取り専用アクセスも許可します。詳細については『Environment Manager ユーザーズ ガイド』を参照してください。

## パースペクティブの詳細

パースペクティブは、ユーザーが利用できる Report Center のレポートおよび機能のセットを定義します。パースペクティブごとの機能は次の表のとおりです。

パースペクティブ	機能
perspective:Basic	テスト > テストの概要 テスト > カテゴリ別エラー テスト > 重要度別エラー 監査 > コード メトリクス > 上位のメトリクス結果 監査 > コード メトリクス > 単一メトリクスの概要
perspective:QA	ダッシュボード > テスター ダッシュボード
perspective:Trends	プラクティス > 静的解析 プラクティス > 単体テスト プラクティス > ソース コードのチェックイン プラクティス > ビルド プラクティス > コード レビュー テスト > 日付別テスト テスト > 変更ベースのテスト セキュリティ > セキュリティ違反

パースペクティブ	機能
perspective:Management	ダッシュボード > プロジェクト ポートフォリオ
perspective:Technical	プラクティス > ツールの使用 テスト > 最近のテスト ログ

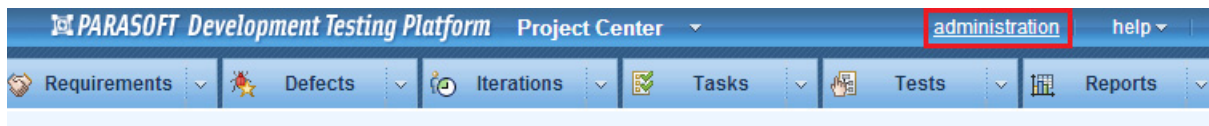
# Project Center 管理ページ

管理者権限を持つユーザーは Project Center の管理ページにアクセスできます。タスクの通知、カスタムのステータスとフィールド、電子署名の要求を始め、さまざまな設定を行うことができます。このセクションでは、Project Center の管理ページで実行できる以下のタスクについて説明します。

- タスクと欠陥の通知メールの設定
- 成果物の名前変更
- カスタム ステータスの追加
- BTS Scanner と RMS Scanner の設定
- カスタム フィールドの使用
- 欠陥のプロパティの指定
- 電子署名と変更理由の設定
- Parasoft Test の設定
- SOAtest サーバーの設定

## Project Center の管理ページを開く

1. 管理ページを開くには、Project Center で、ナビゲーションバーの [ 管理 ] リンクをクリックします。



2. 管理ページを終了するには、ナビゲーションバーの [Project Center] をクリックします。

## タスクと欠陥の通知メールの設定

Project Center を設定して、タスクと欠陥の変更を電子メールで通知することができます。次の 2 種類の通知方法があります。

- **変更ベース**：タスクまたは欠陥が作成あるいは変更されると、電子メールが生成されます。このメールは所有者と監督者に送信されます。
- **日次サマリー**：その日のタスクの変更をまとめた 1 通のメールが毎日零時過ぎに生成されます。このメールは特定のプロジェクト リーダー（そのプロジェクトのマネージャーとして定義されているユーザー）に送信されます。

### 通知を設定する

1. [ 設定 ] メニューの [E-mail] を選択します。
2. [SMTP サーバー設定] で値を入力します。
  - サーバー（必須）

- ポート ( 必須 )
- 接続のセキュリティ ( 必須 )
- ユーザー名 ( 任意 )
- パスワード ( 任意 )
- Sender ( 任意 )
- レルム ( 任意 )

**注意：**" 任意 " のフィールドには、SMTP サーバーが必要とする場合にだけ値を入力してください。

**E-mail Settings**

---

**SMTP Server Settings**

Server:

Port:

Connection Security:

User Name:

Password:

Sender:

Realm:  (Value is required only for those servers that authenticate using SASL realm)

[Test the settings](#)

E-mail address:   (Test message will be sent to the specified e-mail)

---

**E-mail address domain**

User Default Domain:  (When user has no email defined, the email is sent to : login@default\_domain)

---

**E-mail notification**

Send e-mails when Project Center entities (e.g. tasks, defects) are changed, send daily digest e-mail to projects managers

- [E-mail アドレス] フィールドに適切なメール アドレスを入力して [チェック] ボタンをクリックし、SMTP サーバーの設定をテストします。サーバーが正しく動作している場合、指定したメール アドレスにテスト メールが送信されて受信されます。
- ( 任意 ) ユーザーのデフォルト ドメインのメール アドレスを入力します。  
 タスクの変更についての通知は、User Administration モジュールで各ユーザーに設定されたアドレスに基づいてユーザーに送信されます (User Administration モジュールについては 249 ページの「[User Administration] ページ」を参照してください)。  
 DTP でメール アドレスが定義されていないユーザーがいる場合、Project Center はこの管理ページで指定されたデフォルトのドメインアドレスをユーザーのログイン名に追加して、そのアドレスに通知メールを送信しようとします。
- [Project Center タスクが変更されたときに E-mail を送信。プロジェクト マネージャーに日次のタスク サマリー E-mail を送信します。] チェックボックスをオンにします。
- [保存] ボタンをクリックします。



## 成果物の名前変更

DTP では、「成果物」という用語は要求、欠陥、改善、反復、タスク、およびテストを指します。成果物のデフォルトの名前は、できる限り開発手法に中立なものになるように付けられています。定められた用語集や独自の名前付け規則がある場合、この成果物のデフォルト名を変更することができます。

1. [ 設定 ] メニューの [ 成果物の名前付け ] を選択します。
2. 成果物に関連付ける新しい用語を指定します。次の図は、Scrum を採用しているチームの例です。

The screenshot shows the 'Artifact Naming Settings' page. At the top, there are navigation tabs for 'Main', 'Projects', and 'Settings'. Below the tabs, the page title is 'Artifact Naming Settings'. The main content area is divided into four sections, each representing a different artifact type:

- Defect:** Customized Name: Bug, Customized Short Name: Bug
- Enhancement:** Customized Name: Feature, Customized Short Name: Feature
- Iteration:** Customized Name: Sprint, Customized Short Name: Sprint
- Requirement:** Customized Name: User Story, Customized Short Name: US

3. [ 保存 ] をクリックします。

Project Center と Report Center の両方で、指定した名前がその成果物に割り当てられます

Project Center のデフォルトの成果物名に戻すには、[ デフォルトに設定 ] をクリックします。

## カスタム ステータスの追加

DTP では、欠陥、シナリオ、シナリオの実行、およびテスト セッションに対してチームが使用するカスタム ステータスを含めることができます。カスタム ステータスを使って、関連する成果物を分類、発見、およびレビューすることができます。

1. [ 設定 ] > [ 成果物のステータス ] を選択します。
2. [ 新規 ] をクリックし、新しいステータスの名前を入力します。

Defect Statuses

5 Item(s) New

Active	Name	State
<input checked="" type="checkbox"/>	New	Unresolved - Editable
<input checked="" type="checkbox"/>	Assigned	Unresolved - Editable
<input checked="" type="checkbox"/>	Resolved	Resolved - Editable
<input checked="" type="checkbox"/>	Verified	Resolved - Read-only
<input checked="" type="checkbox"/>	New status	Unresolved - Editable

3. 成果物のステータスを指定します。編集可能ステータスを設定することで、成果物の他のプロパティも更新することができます（ランク、オーナー、期日など）。次の表は、成果物に設定できるステータスについての説明です。

成果物のステータス	利用できるステータス	定義	例
欠陥のステータス	未解決 - 編集可能	欠陥は未解決であり、そのプロパティは更新できます。	New
	未解決 - 読み取り専用	欠陥は未解決であり、そのプロパティは更新できません。	Assigned
	解決済み - 編集可能	欠陥は解決済みであり、そのプロパティは更新できます。	Resolved Fixed
	解決済み - 読み取り専用	欠陥は解決済みであり、そのプロパティは更新できません。	Verified Closed
要件のステータス	未解決 - 編集可能	要件は未解決であり、そのプロパティは更新できます。	Pending Review
	未解決 - 読み取り専用	要件は未解決であり、そのプロパティは更新できません。	Draft
	解決済み - 編集可能	要件は解決済みであり、そのプロパティは更新できます。	Pending Review
	解決済み - 読み取り専用	要件は解決済みであり、そのプロパティは更新できません。	Completed
シナリオのステータス	編集可能	シナリオは更新できます。	To Review
	読み取り専用	シナリオは更新できません。	Approved
シナリオ実行のステータス	編集可能	シナリオ実行は更新できます。	To Review
	読み取り専用	シナリオ実行は更新できません。	Approved

テストセッションのステータス	実行可能 - 編集可能	テストセッションは実行することができ、そのプロパティは更新できます。	Draft
	実行可能 - 読み取り専用	テストセッションは実行することができ、そのプロパティは更新できません。	Template
	実行不可能 - 編集可能	テストセッションは実行不可能であり、そのプロパティは更新できません。	Code Pending
	実行不可能 - 読み取り専用	テストセッションは実行不可能であり、そのプロパティは更新できません。	Legacy

4. このページの下部の [ 保存 ] ボタンをクリックします。

---

## BTS Scanner と RMS Scanner の設定

DTP は、GUI を介して、または \$DTP\_HOME/grs/config/bts ディレクトリの統合用 XML ファイルを設定することで、バグ トラッキング システムまたは要求管理システムと統合することができます。詳細については 336 ページの「バグ トラッキング システムおよび要求管理システムの統合」を参照してください。

---

## 欠陥のプロパティの指定

DTP の内部的な欠陥トラッキング システムを使用している場合、欠陥のオーナーと監督者のデフォルトを設定できます。

1. [ 設定 ] > [ 欠陥のプロパティ ] を選択します。
2. [ 編集するフィールドの選択 ] ドロップダウン メニューをクリックして欠陥フィールドを選択します。

3. 値を入力して [ 新規 ] をクリックし、選択したフィールドに値を追加します。

Defect/Enhancement Fields Values Configuration

Project  
Default Project

Choose field to edit  
Component

Edited field: Component

1 Item(s) New Component Property New

Name	Owner	Observer(s)
Old Component Property		

## カスタム フィールドの使用

DTP では、要求、欠陥、シナリオ、およびテスト セッションを分類するためにチームが使用するカスタム フィールドを含めることができます。ここで入力したカスタム フィールドは、関連する成果物を分類、発見、およびレビューするために使用できます。

## カスタム フィールドを追加する

新しいカスタム フィールドを追加するには、次の操作を行います。

1. [ 設定 ] > [ カスタム フィールド ] を選択します。
2. ステータスを追加する成果物（たとえば欠陥、要求など）のタブで [ 新規 ] をクリックします。そしてテキスト フィールドの場合は [ テキスト入力 ] を、コンボボックスの場合は [ 単一選択リスト ] を選択します。


Custom Fields Settings

Requirement Defect Task Scenario Test Session

5 Item(s) New

Name	Field Type	Default Value	Built-in	Active	Unique Name
Parent	Text Input		true	true	tParent
Predecessor(s)	Text Input		true	true	tPredec
Type	Single Select List	Development	true	true	tType
task_custom1	Text Input		false	true	cf10
task_custom2	Text Input		false	true	cf11

3. カスタム フィールドの設定を指定します。
  - q. [ラベル] フィールドにフィールド名を入力します。
  - r. このフィールドについての説明を入力します。
  - s. フィールドをアクティブにするかどうか (ユーザーに見えるようにするかどうか) を指定します。
4. [保存] をクリックするか、次のステップに進んで単一選択リストのカスタム フィールドを設定します。
5. [利用可能な値] の表に値を入力し、[新規] をクリックします。このステップで入力した値はコンボボックスで使用できます。

 General

Label: \*

Description:


Active:

Value Definition

Default Value:

Available Values

1 Item(s)

	Unique Name	Built-in	Name	
		true	--Not specified--	▲ ▼

6. 上下の矢印を使って、コンボボックスの値の順序を変更します。
7. 新しいカスタム フィールドのデフォルト値を入力します。
8. [保存] をクリックします。

## フィールドをアクティブ化 / 非アクティブ化する

フィールドを非アクティブにすると、そのフィールドは UI から削除されます (フィールドそのものは削除されません)。管理ページではフィールドを再びアクティブにすることができます。

1. [設定] > [カスタム フィールド] を選択し、成果物のタブをクリックします。ページ下部の [アクティブなカスタム フィールドだけを表示] を有効にすると、非アクティブのフィールドを非表示にすることができます。


2. 編集アイコンをクリックします。

Main Projects Settings

Custom Fields Settings

Requirement Defect Task Scenario Test Session

1 Item(s) New

	Label	Field Type	Default Value	Built-in	Active	Unique Name
 Custom 1A Edit field definition		Text Input		false	true	cf4

3. [アクティブ] チェックボックスをオン/オフにして、UI に対してフィールドを追加 / 削除します。

General

Label: \* Custom 1A

Description:

Active:

Value Definition

Default Value: ?

## 欠陥のプロパティの設定









主要な欠陥フィールドで使用できる値を指定するには、欠陥のプロパティを設定します。たとえば、[検出バージョン] フィールドに値として 1.0、4.5、4.6、4.7 が表示されるように設定できます。

Choose field to edit

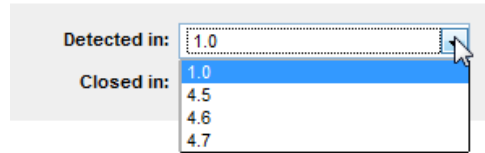
Detected in

Edited field: Detected in

4 Item(s) New

	Name	
	1.0	
	4.5	
	4.6	
	4.7	

ユーザーが欠陥を入力するとき、[ 検出バージョン ] ドロップダウン リストが次のように表示されます。



## 電子署名と変更理由の設定

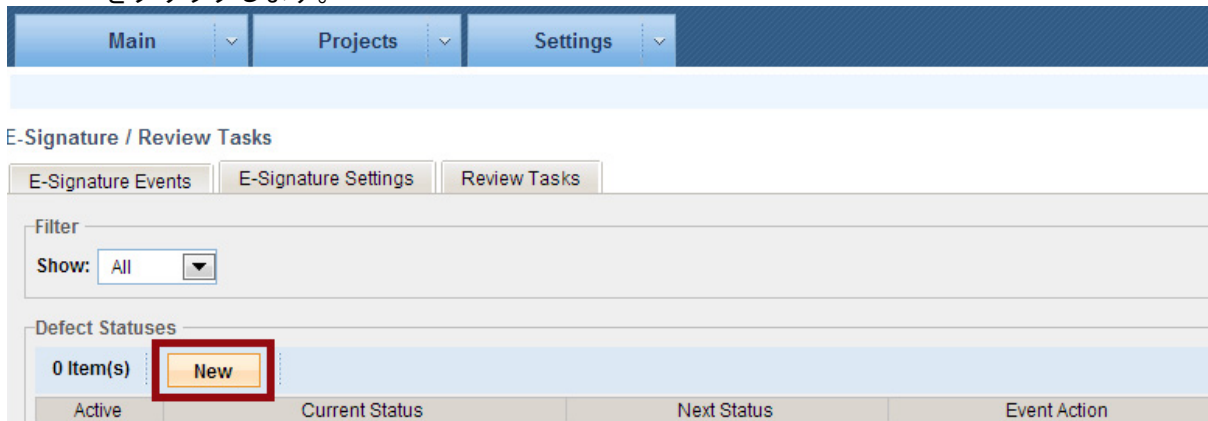
あるステータスから別のステータスに項目を変更するときに、必ず電子署名または変更理由を入力しなければいけないように Project Center を設定できます。たとえば、項目のステータスが "ドラフト" から "承認済み" に変わるときに電子署名を要求できます。入力された電子署名または変更理由は [ 変更履歴 ] タブに記録されます (75 ページの「電子署名 / 変更理由の入力とトラッキング」を参照)。

電子署名と変更理由の設定は、要求、欠陥、シナリオ、シナリオの実行、およびテスト セッションに対して指定できます。

## 電子署名または変更理由を追加する

新しい電子署名または変更理由を追加するには、次の操作を行います。

1. [ 管理 ] リンクをクリックして管理ページを開き、[ 設定 ] > [ 電子署名 / 通知の設定 ] を選択します。
2. 電子署名を追加する成果物 (たとえば欠陥、シナリオなど) のセクションで [ 新規 ] ボタンをクリックします。



3. [現在のステータス] および [次のステータス] で、いつ電子署名を要求するかを指定します。

E-Signature / Review Tasks

E-Signature Events | E-Signature Settings | Review Tasks

Filter  
Show: All

Defect Statuses  
0 Item(s) [New](#)

Active	Current Status	Next Status	Event Action
<input checked="" type="checkbox"/>	New	New	E-Signature

Requirement Statuses  
0 Item(s) [New](#)

Active	Current Status	Next Status	Event Action
NO DATA TO DISPLAY			

4. [イベント アクション] フィールドで、電子署名を要求するか変更理由を要求するかを指定します。
5. [保存] ボタンをクリックします。

## 電子署名のメッセージを設定する

カスタムの電子署名メッセージを作成する方法については、76 ページの「カスタム電子署名」を参照してください。

## 電子署名の設定をアクティブ化 / 非アクティブ化する

電子署名の設定を非アクティブにすると、その電子署名は UI から削除されます。ただし、電子署名そのものは保持されるので、後で再び設定をアクティブにすることができます。

- [アクティブ] チェックボックスをオン / オフにして、UI に対して電子署名の設定を追加 / 削除します。
- ページ下部の [アクティブなカスタム フィールドだけを表示] を有効にすると、非アクティブの電子署名の設定を非表示にすることができます。

## ステータス変更のレビュー タスクの設定

項目のステータスが変化したときにレビュー タスクを自動生成するように Project Center を設定できます。また、通知するユーザーグループや、自動生成されるレビュー タスクに割り当てるために使用するタスク作成ワークフローを指定することもできます。

たとえば、欠陥のステータスが "進行中" から "修正済み" に変わったときにレビュー タスクを作成するように DTP を設定することができます。さらに、タスク作成ワークフローをラウンド ロビンに設定することで、欠陥のステータスが初めて "進行中" から "修正済み" に変わったときに、レビュー タスクが生成されて QA チームのメンバーに割り当てることができます。次回、欠陥の変更ステータスが "進行中" から "修正済み" に変わると、レビュー タスクが生成されて別のチームメンバーに割り当てられます。この "ラウンドロビン" の割り当ては、チーム メンバー全員が 1 個のレビュー タスクを受け取るまで続き、その後ワークフローが繰り返されます。



DTP は、欠陥、シナリオ、シナリオ実行、およびテスト セッションのステータスが変化したときにレビュー タスクを作成するように設定することができます。

## レビュー タスクを設定する

1. Project Center で、ナビゲーション バーの [ 管理 ] リンクをクリックし、[ 設定 ] > [ 電子署名 / 通知の設定 ] を選択します。
2. [ レビュー タスク ] タブをクリックし、[ 新規 ] をクリックします。レビュー タスクを自動生成するには、電子署名を定義する必要があります。
3. [ イベント ] 列で、いつレビュー タスクを生成するかを指定します。たとえば、成果物のステータスが "New" から "Verified" に変わったときにレビュー タスクを生成する場合、次の図のように設定します。

Defect Statuses		
Event	Review Task Owner - Member of Group	Create Tasks Workflow
New->Verified	Adamt team	Round Robin

4. [ レビュータスクオーナー - グループのメンバー ] からユーザー グループを指定します。このユーザー グループは、自動生成されたレビュー タスクを割り当てるチーム メンバーが属するグループです。このリストには、[User Administration] ページで指定されたすべてのユーザー グループが表示されます。
5. [ タスク生成ワークフロー ] フィールドで、ユーザー グループ メンバーにタスクを割り当てるために使用するワークフローを指定します。
  - Round Robin ワークフローでは、ステータスが変化したときに、グループ中の一人のメンバーに対してレビュー タスクが作成されます。
  - Notify All ワークフローでは、ステータスが変化したときに、グループ中の対応できるメンバー全員に対してレビュー タスクが作成されます。
6. [ 保存 ] ボタンをクリックします。

各項目タイプ (欠陥や要求など) のステータスが変化したときに、特定のチーム メンバーに対してレビュー タスクが自動的に作成されます。このレビュー タスクには "Review [artifact type] : [artifact ID] [artifact name]" の書式で名前が付けられます。[Description and Comments] の部分には「Status changed from [status 1]New to [status 1]. Click here to review: [artifact link].」という文章が自動的に挿入されます。

例 :

Project: *	JustTestProject
Name: *	Review Defect: [92,740] My defect here

### Description and Comments

#### Description

Status changed from New to Assigned.

Click here to review: [PR 92740](#)

## JMS イベント ブローカー ポートの設定

JMS イベント ブローカーは、DTP で発生したイベントについてリアルタイムの情報を提供します。デフォルト ポートは 61617 です。ただし、このポートは Report Center または Project Center の管理者ページから設定することができます。

1. [ 設定 ] > [ イベント ブローカーの構成 ] を選択し、[ 停止 ] をクリックします。
2. 新しいポート番号を入力して [ 保存して再開 ] をクリックします。

現在サブスクリプションで利用できる機能を含め、JMS イベント ブローカーの詳細については 853 ページの「DTP イベントのサブスクリプション」を参照してください。

## Parasoft Test の設定

「すべてのプロジェクト」のテストを実行するために Parasoft Test 製品 (C++test、dotTEST、SOAtest、Jtest) を使用するとき、このページで設定した localsettings が使用されます。「特定のプロジェクト」に対する localsettings の設定については 237 ページの「Parasoft Test の設定」を参照してください。

Parasoft Test ツールごとにファイルを作成する代わりに、このページで一度に localsettings を設定できます。この設定は、Parasoft Test 製品が DTP プロジェクトに接続するときに、自動的にその Parasoft Test 製品に伝播されます。

例：

```
#Report Center Settings
grs.data.port=32323
#License Settings
license.network.host=ntp.company.com
license.network.port=2222
license.use_network=true
#Mail Settings
report.mail.domain=company.com
```

```

report.mail.from=john.doe
report.mail.password=123456789
report.mail.server=mail.company.com
report.mail.username=john
#Team Server Settings
tcm.server.accountLogin=true
tcm.server.enabled=true
tcm.server.name=ntp.company.com
tcm.server.password=123456789
tcm.server.port=18888
tcm.server.username=team_user

```

下記の値が定義されていない場合、DTP は自動的に値を設定します。Data Collector や Mail Server のポートなどのように、DTP の全般的な構成に基づいて値が設定されます。

```

#Report Center Settings
grs.data.port
#License Settings
license.network.host
license.network.port
#Mail Settings
report.mail.domain
report.mail.from
report.mail.password
report.mail.server
report.mail.username
#Team Server Settings
tcm.server.name
tcm.server.port

```

自動的に設定された値は、ユーザーが手動で値を指定したときに上書きされます。

詳細については『Parasoft Test ユーザーズ ガイド』を参照してください。

---

## SOAtest サーバーの設定

Project Center のインターフェイスから (SOAtest サーバーの Web サービスを介して) チームメンバーが SOAtest .tst ファイルを実行できるように設定できます。次の操作を行います。なお、現在 DTP は、アクセス制御が必要な場合、SOAtest サーバーへのアクセスをサポートしていません。DTP が SOAtest サーバーにアクセスする必要がある場合、アクセス制御が無効化された SOAtest サーバーを使用する必要があります。

1. [SOAtest Server の URL] で、SOAtest サーバーの Web サービス インターフェイスのサーバーとポート番号を入力します。

2. [SOAtest Test の設定] で、Project Center インターフェイスから SOAtest テストを実行するときにデフォルトで使用するテスト コンフィギュレーションを指定します。

**SOAtest Server Settings**

---

**SOAtest Server Settings**

SOAtest Server URL

Server:


Port:

SOAtest Test Settings


Default Test Configuration:

3. [保存] をクリックします。

これらの設定が SOAtest テスト タイプのデフォルトとして設定される点に注意してください。たとえば、あるテストで別のテスト コンフィギュレーションが必要な場合などには、必要に応じてこれらの設定をテスト レベルで変更できます。

 New Scenario

Scenario

 Scenario Steps

Step #1: Invoke SOAtest

SOAtest Server Parameters

Host:

Port:

SOAtest Test Suite Parameters

Path:

Configuration:

# Data Collector の構成

組織のデータ要件に合わせて、Data Collector のデフォルトのストレージおよびアップロードの制限値を変更できます。

## レポート ストレージ閾値の変更

Data Collector がアナライザーからファイルを受信して処理する際、処理されたレポート ファイルは `[DTP_HOME]/data/_DEFAULT_/dc/stored` フォルダに置かれます。このフォルダには、`YEAR/MONTH/DAY/XXX/YYYY` (`XXX` および `YYY` はファイル名を基に計算された整数値です) という書式のサブフォルダがあります。

保存されるフォルダの最大サイズを設定し、DTP が古いエントリを自動的に削除するよう指定できます。

1. `[DTP_HOME]/grs/config/` ディレクトリの `DCCConfig.xml` ファイルを開き、次のエントリを探します。

```
<stored-folder>
  <max-size>20</max-size>
  <size-to-clean>5</size-to-clean>
</stored-folder>
```

2. `<max-size>` プロパティの値 (単位: GB) を許される最大のストレージ容量に変更します。デフォルトは 20 GB です。
3. `<size-to-clean>` プロパティの値 (単位: GB) を最大容量に達した時に削除するデータの量に変更します。デフォルトは 5 GB です。
4. ファイルを保存します。

`DCCConfig.xml` ファイルにエントリがない場合、デフォルト値が使用されます。

## アップロードされるレポートの最大サイズの変更

デフォルトでは、Data Collector が許容する XML レポート ファイルの最大サイズは 512 M です。次の Data Collector JVM 引数を追加すると、制限値を変更できます。

```
-Dcom.parasoft.sdm.api.rawstorage.datacollector.uploadMaxSize=512
```

Linux では、`[DTP_HOME]/bin/variables` ファイルにある次の文字列を変更します。

```
JAVA_DC_CONFIG_ARGS=" -
Dcom.parasoft.sdm.api.rawstorage.datacollector.uploadMaxSize=512"
```

---

# Report Center と Project Center のコンポーネントのポート変更

このセクションでは次のタスクの概要について説明します。

- Web サーバーのポート番号の変更
- Data Collector のポート番号の変更

---

## Web サーバーのポート番号の変更

DTP Server のデフォルト ポートは、Windows が 80、Linux/Solaris が 8080 です。ただし、server.xml ファイルを編集してポートを変更することができます。DTP Server のポートを変更する前に、必ず『Development Testing Platform クイック インストール ガイド』の Tomcat サーバーについての情報を参照してください。

### デフォルトのポート番号を変更する

1. DTP\_HOME/tomcat/conf/server.xml ファイルを開きます。
2. デフォルトのポート番号を変更します。次の例は Windows の場合です。  
変更前：  

```
<Service name="PST">  
<Connector port="80"
```

  
変更後：  

```
<Service name="PST">  
<Connector port="8888"
```
3. ファイルを保存します。
4. Parasoft DTP サービスを再起動します。
5. ブラウザーで、変更したポート番号を追加して URL を入力します。たとえばこの例では、  
http://dtp.company.com:8888 のように入力します。

---

## Data Collector のポート番号の変更

Report Center と Project Center は、必要なデータ形式をサポートするソフトウェア製品からのデータを受け取ることができます。Parasoft 社のテスト ツールである Jtest、C++test、dotTEST などが該当します。テスト ツールのデータは Data Collector に送られます。Data Collector はデフォルトでポート番号 32323 をリスンします。

デフォルト ポートを変更する必要がある場合、次の操作を行います。

1. xml ノードでポート番号を変更して DTP\_HOME/grs/config/DCCconfig.xml ファイルを変更します。

```
<data-collector-port>32323</data-collector-port>
```

2. ファイルを保存します。
3. Data Collector を再起動します。

# Cache Report Executor の設定

Cache Report Executor のジョブは、staticLinksConfig.xml 構成ファイルで定義されたレポートを参照し、キャッシュします。デフォルトでは、このジョブは午前 5 時 (allowFromHour=5) に開始するように設定されています。デフォルト設定がニーズに合わない場合、Cache Report Executor ジョブを変更して、キャッシュ開始時刻とキャッシュ対象のレポート リストを変更することができます。

Cache Report Executor について問題があった場合は、418 ページの「Cache Report Executor」を参照してください。

## キャッシュ開始時刻の設定

キャッシュ開始時刻のスケジュールは、CronConfig.xml ファイルの Job id="Cache Report Executor" xml 要素で設定することができます。

## キャッシュするレポート リストの設定

Cache Report Executor のジョブが実行されたときにキャッシュするレポートのリストを設定するには、staticLinksConfig.xml ファイルを編集します。次はこのファイルの例です。

```
<?xml version="1.0" encoding="UTF-8"?>
<reports>
  <report descr="Architect Dashboard" xreport-id="architect_dashboard/
composite_desc" composite="true" params=""/>
  <report descr="Practices > Coding Standards" xreport-id="practices/
cs_composite_desc" composite="true" params=""/>
  <report descr="Practices > White Box" xreport-id="practices/wb-bb/
wb_composite_desc" composite="true" params=""/>
  <report descr="Practices > White Box > Coverage" xreport-id="practices/
wb-bb/WBCoverageDetails" composite="false"
params="period=10drops&dateMode=period"/>
  <report descr="Practices > Black Box" xreport-id="practices/wb-bb/
bb_composite_desc" composite="true" params=""/>
  <report descr="Practices > Black Box > Coverage" xreport-id="practices/
wb-bb/BBCoverageDetails" composite="false"
params="period=10drops&dateMode=period"/>
  <report descr="Audit > Drop Grade" xreport-id="audit/DropGrade"
composite="false" params="period=10drops&dateMode=period"/>
  <report descr="Audit > Errors By Category" xreport-id="errors/
ErrorsByCategory" composite="false"
params="period=10drops&dateMode=period"/>
  <report descr="Audit > Errors By Severity" xreport-id="errors/
ErrorsBySeverity" composite="false"
params="period=10drops&dateMode=period"/>
</reports>
```



次の表は、上記のファイルに含まれるタブの一部とその説明です。

タグ	説明
reports	ルート要素。
report	キャッシュする 1 つのレポートについての情報を格納します。
descr	キャッシュするレポートについての簡単な説明。
xreport-id	キャッシュするレポートを特定する xreport 識別子を格納します。
composite	レポートが複合レポートの場合、この属性を true に設定します。そうでない場合、false に設定します。
params	レポートをキャッシュするために使用されるレポート パラメータを持つ属性。アンパサンド (&) を使って複数のパラメータを連結します。

## キャッシュするレポートの追加

xreport-id 属性は、適切なレポートの URL をコピーして設定することができます。例としてアーキテクト ダッシュボードを使って説明します。

1. Report Center の Web ページに移動します。

```
http://localhost/grs
```

2. キャッシュするレポートに移動します。たとえば：

```
http://localhost/grs/
xarchitect_dashboard.jsp?xreportcomposite=architect_dashboard/
composite_desc.
```

3. リンクをコピーします。この例では architect\_dashboard/composite\_desc です。これを xreport-id 属性の値として貼り付けます。
4. composite 属性が true であることを確認します。

## Report Center とレポート キャッシュのメカニズム

Report Center のレポートは次の 2 つの方法でキャッシュされます。

- ユーザーがブラウザしてレポートを生成したとき、レポートはキャッシュに格納されます。
- 毎晩 Report Center のバックグラウンド ジョブが実行するときに、`ntp\grs\config\staticLinksConfig.xml` 構成ファイルで指定されたレポートがキャッシュされます。

レポート キャッシュは次の 3 つの方法で無効化できます。

- Report Center バックグラウンド ジョブの夜間 (夜の 12 時以降) の実行。起動時刻は `ntp\grs\config\CronConfig.xml` ファイルの `CleanCacheReportExecutor` で設定でき

ます。夜間ジョブの構成の詳細については「レポート キャッシュのカスタマイズ」を参照してください。

- Report Center の管理者モードで、[ ツール ] メニューの [ データ キャッシュ ] > [ キャッシュの無効化 ] を選択します。
- DTP Server を再起動します。

#### 注意!

- 特定のレポートがキャッシュされ、Parasoft ツールからの新しいログを Data Collector が受け取るとき、このレポート キャッシュは無効化されません。レポートはキャッシュから表示されます。そのため、新しいログをレポートで参照するには、キャッシュをメニューから無効化するべきです。そうしないと、レポート キャッシュが午前零時過ぎに自動的に実行されます。
- レポートのキャッシュ データは dtp/reportcenter/datacache ディレクトリに格納されます。特定のパラメーターを持つレポートごとに、2 種類のファイルが格納されます。1 つは .dat ファイルであり、もう 1 つは .par ファイルです。

## レポート キャッシュのカスタマイズ

レポート キャッシュは次の 2 箇所カスタマイズすることができます。

- dtp/grs/config/CronConfig.xml 構成ファイル。このファイルは Report Center のバックグラウンド ジョブの設定を定義します。  
キャッシュに関連するのは次の 2 つのジョブです。
  - CacheReportExecutor:  
dtp\reportcenter\staticLinksConfig.xml 構成ファイルで定義されたレポートをブラウザしてキャッシュするバックグラウンド ジョブ。このジョブはデフォルトでは毎朝 5 時に実行されます。
  - CleanCacheReportExecutor:  
レポート キャッシュを無効化するジョブ。デフォルトでは、毎晩 12:01 に実行されます。
- Report Center の管理者ページで [ ツール ] メニューの [ データ キャッシュ ] > [ Cache Settings ] を選択します。
  1. キャッシュのオン / オフを切り替えます。
  2. Report サーバーのアイドル時間を分で指定します。レポート キャッシュを自動生成する CacheReportExecutor は、停止した後にこのアイドル時間が過ぎると、再び動作し始めます。

CacheReportExecutor は CronConfig.xml で定義された時刻 ( デフォルトは allowFromHour=5 a.m. ) に開始し、次のいずれかの状況が発生するまで動作します。

- staticLinksConfig.xml で指定されたすべてのレポートがキャッシュされた場合。
- CacheReportExecutor が動作中にユーザーが Report Center レポートを参照し始め、CacheReportExecutor が一時停止し、すべてのユーザーが Report Center レポートの参照を終了した後に、Report サーバーのアイドル時間が経過して CacheReportExecutor が再び動作した場合。

- CronConfig.xml で定義された allowToHour に CacheReportExecutor が達した場合 (staticLinksConfig.xml で定義されたすべてのレポートがキャッシュされたとき、それまでに停止していなければ、この時点で CacheReportExecutor は停止します)。

# Team Server の設定

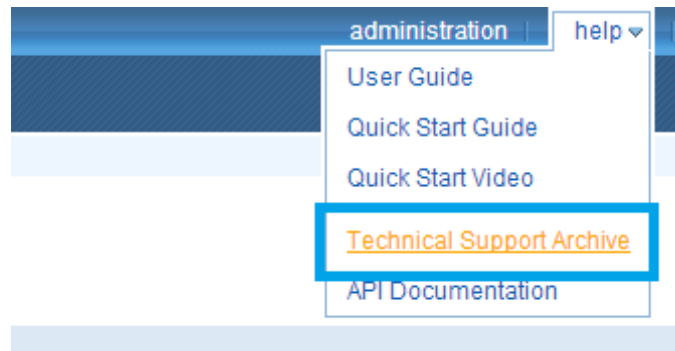
このセクションの内容：

- 一般的な設定
- データ管理
- サポート

## テクニカル サポート アーカイブの作成

問題が発生した場合、ログのパッケージを作成して Parasoft テクニカル サポートに送付することができます。

Project Center で、[ ヘルプ ] > [ テクニカル サポート アーカイブ ] を選択すると、DTP のログが格納されたフォルダーの zip ファイルをダウンロードできます。



---

# テクニカル サポート

テクマトリックス株式会社では、ユーザーの方々に対し、きめ細かなサポート サービスを行っています。Parasoft ツールのご使用中にトラブルや疑問が生じた場合は弊社テクニカル サポートにご連絡ください。

テクマトリックス株式会社

Parasoft 製品テクニカル サポート センター

Email: [parasoft\\_support@techmatrix.co.jp](mailto:parasoft_support@techmatrix.co.jp)

# License Server の使用

このセクションの内容：

- License Server の設定
- ライセンスの管理

# License Server の設定

このセクションの内容：

- License Server の設定
- 権限の設定

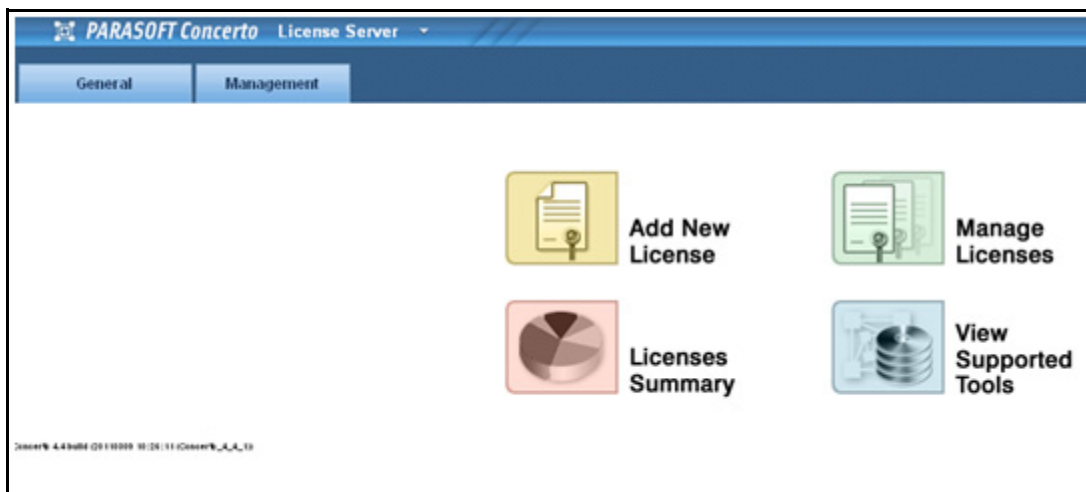


# License Server の設定

License Server を設定するには、次の手順に従います。

1. License Server ホーム ページを開きます ( 図 )。

License Server ホーム ページ



2. [General] メニューの [License Server Configuration] をクリックします。

License Server Configuration ページが開きます。

## License Server Configuration ページ

### License Server configuration

[\[Back\]](#)

**Note:** You need to restart License Server for port change to take effect.

License Server port:  [\[Restart License Server\]](#)

Clients should start deactivated:

Concurrent connections limit:

---

### Access logs management

Log cleaning

Remove log files older than  days.

Sending to Report Center

Host:

Port:

Log send time:  :

Last sent date: 2011-02-23 [\[Reset date\]](#)

[\[Send logs to Report Center\]](#)

Use log files from location:

**Note:** It is not recommended to change the default path.

default (/home/servers/parasoft/concerto/logs)

[\[Save\]](#)

3. 必要に応じて、以下のフィールドに値を入力します。

- **License Server Port** インストールの後、デフォルトでは 2002 が設定されています。必要に応じてポート番号を変更してください。変更した後はサービスを再起動する必要があります。
- **Clients should start deactivated** オンの場合、Parasoft ツールの "Start deactivated" オプションを強制的に有効にします。

- **Concurrent connections limit** 同時に処理できる、ツールのリクエスト (License Server への接続) の最大数を設定します。デフォルトは 100 です。
- **Log cleaning** チェックボックスをオンにし、古いログがデータベースから削除されるまでの日数を入力します。
- **Sending to Report Center** 選択すると License Server アクセス ログが Report Center に送信されます。次のオプションを指定します。
  - **Host** Report Center ホスト。
  - **Port** Report Center ポート。
  - **Log send time** 毎日 License Server が Report Center にログを送信する時刻。
  - [Send logs to Report Center] このリンクをクリックすると、いつでも Report Center にログを送信できます。
- **Use log files from the location** アクセス ログの格納場所を指定します。デフォルトのパスの使用を推奨します。ログ ファイル アクセスの場所を変更した場合、そのパスを入力します。この設定に従って、License Server はログを削除したり Report Center にデータを送信します。

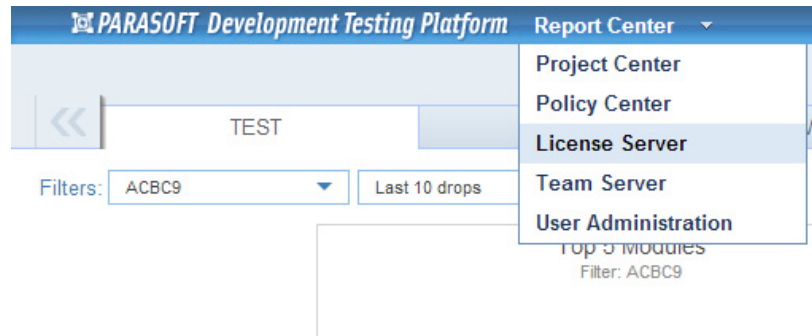
# ライセンスの管理

このセクションの内容：

- ライセンスの追加と管理
- ライセンス サマリ
- サポートされているツールの表示
- アクセス ログの表示

## ライセンスの追加と管理

License Server は、Development Testing Platform (DTP) と Parasoft コード解析ツールを含め、Parasoft ライセンスへのローカル エリア ネットワーク アクセスを管理することができます。適切な権限がある場合、License Server で新しいライセンスを追加したり既存のライセンスを管理することができます。License Server を表示するには、DTP のツール スイッチャーから License Server を選択します。



### ライセンスの追加

各 Parasoft 製品のライセンスは、License Server を実行中のサーバーのマシン ID に対して生成する必要があります。詳細については、Parasoft 担当者に問い合わせてください。

1. License Server ページを開き、[Add New License] をクリックします。
2. 次のいずれかの操作を行います。
  - Add New License フォームでライセンス パラメーターを入力します。
  - 入手した完全なライセンス パスワードをコピーして貼り付けます。パスワードは組織の Parasoft 担当者から電子メールで送信されます。

3. [Add new license] をクリックします。

**Add New License**

[\[Back\]](#)

Machine ID: WIN32-4e738235

Fill out license parameters

Tool:

Version:

Architecture:

Expiration date:

Password:

[\[Add license\]](#)

Or cut and paste license password lines here as they were sent to you :  
(example : `Insure++.password 5.1 SOL-80c0a8b6 6488360 41ae382464146e88`)

[\[Add license\]](#)

## ライセンス パラメーター

**Tool:** Parasoft ツール (たとえば dotTEST、C++test、SOAtest、Static Analysis Engine など)。

**Version Number:** バージョン番号にはツールの [バージョン情報] パネルからアクセスできます。

**Architecture:** ライセンスが割り当てられた正しいアーキテクチャを選択する必要があります (たとえば LINUX2 など、マシン ID の最初の部分です)。

**Expiration date:** Parasoft 担当者から入手した有効期限コードを入力します、これは、ライセンス サーバーが有効期限切れになる日付またはライセンスが有効な期間です。mm/dd/yyyy または mm/dd/yy の書式を使用します。

**Password:** ライセンスのパスワード コード。

追加されたライセンスについての情報は Installed Tool Licenses List ページで参照できます。このライセンスについてのエントリは psrc.xml ファイル (ライセンス情報を記録するファイル) に追加されます。psrc.xml ファイルは License Server のインストール ディレクトリにあります。

## 現在インストールされているライセンスの管理

現在インストールされているライセンスの一覧を表示するには、License Server ホーム ページを開いて [Manage Licenses] をクリックします。

Installed Tool Licenses List					
<a href="#">[Back]</a>					
	Tool name	Machine ID	Expiration date	Available tokens	
1	<a href="#">C++Test 6.0</a>	LINUX2-ab85616	Unlimited	Unlimited	<a href="#">[Remove]</a>
2	<a href="#">C++Test 7.0</a>	UNI-ab85616	Unlimited	1	<a href="#">[Remove]</a>
3	<a href="#">jtest! 7.0</a>	UNI-ab85616	Unlimited	3	<a href="#">[Remove]</a>
4	<a href="#">jtest! 8.0</a>	UNI-ab85616	03 Oct 2007	1	<a href="#">[Remove]</a>
5	<a href="#">jtest! 8.0</a>	UNI-ab85616	03 Oct 2007	1	<a href="#">[Remove]</a>
<b>Not linked upgrade licenses:</b>					
	Tool name	Machine ID	Expiration date	Available tokens	
1	<a href="#">jtest! 8.0</a>	UNI-ab85616	Unlimited	3	<a href="#">[Link to]</a> <a href="#">[Remove]</a>

次のライセンス管理タスクを行うことができます。

- ライセンスの詳細の表示
- ライセンスのリンク
- ライセンスの予約
- ライセンスの削除

### ライセンスの詳細の表示

ライセンスの詳細を表示するには、Installed Tool Licenses List でサポートされているツール名のリンクをクリックします。

- ツール
- マシン ID
- 有効期限
- ライセンス (使用中、利用可能、すべて)
- ライセンスで許可された機能 (ライセンスで許可されたすべての機能)

### ライセンスのリンク

アップグレード ライセンスは、新しいバージョンへの承認されたアクセスを提供することによって、メイン ライセンスのアクセス権を拡張します。その上限は、特定ツールのメイン ライセンス内で利用できるトークンの数によって決定されます。ライセンスをリンクすると、メイン ライセンスのトークンとアップグレードのトークンを区別なく使用することができます。

たとえば、ある製品のライセンス（トークン）を 5 個購入し、徐々に新しいバージョンにアップグレードする場合、古い製品バージョンと新しい製品バージョンをリンクすることができます。この場合、たとえば新しいバージョンで 3 ライセンスを使用し、古いバージョンで 2 ライセンスを使用できます。また、5 ライセンスすべてをどちらかのバージョンで使用することもできます。

アップグレード ライセンスをメイン ライセンスにリンクするべき理由として、もう 1 つ、アップグレード ライセンスはメイン ライセンス内でしか予約できないことがあります。メイン ライセンスのトークンを予約することによって、ユーザーはニーズに合わせてメイン トークンまたはアップグレード トークンのいずれかを使用することができます。ライセンスの予約の詳細については 308 ページの「ライセンスの予約」を参照してください。利用可能なアップグレード ライセンスは、Manage Licenses ページに表示されます。

Not linked upgrade licenses:					
	Tool name	Machine ID	Expiration date	Available tokens	
1	jtest! 8.0	UNI-ab85616	Unlimited	3	<a href="#">[Link to]</a> <a href="#">[Remove]</a>

1. [\[Link to\]](#) リンクをクリックします。

Linking upgrade license ページが表示されます。

**Linking upgrade license**  
[\[Back\]](#) [\[Refresh page\]](#)

<p><b>Select the license from the following list:</b></p> <p><b>Tool:</b> jtest! 7.0 <a href="#">[Select]</a></p> <p><b>Machine ID:</b> UNI-ab85616</p> <p><b>Expiration date:</b> Unlimited</p> <p><b>Tokens:</b> 3</p> <p><b>License features:</b></p> <ul style="list-style-type: none"> <li>◆ jtest!</li> <li>◆ jtest! Coding Standards</li> <li>◆ jtest! Command Line Mode</li> <li>◆ jtest! RuleWizard</li> <li>◆ jtest! Security</li> <li>◆ jtest! Test case parametrization</li> <li>◆ jtest! Test case sniffer</li> <li>◆ jtest! Unit Testing</li> </ul>	<p><b>Upgrade license:</b></p> <p><b>Tool:</b> jtest! 8.0 <a href="#">[Clear]</a></p> <p><b>Machine ID:</b> UNI-ab85616</p> <p><b>Expiration date:</b> Unlimited</p> <p><b>Tokens:</b> 3</p> <p><b>License features:</b></p> <ul style="list-style-type: none"> <li>◆ jtest!</li> <li>◆ jtest! BugDetective</li> <li>◆ jtest! Coding Standards</li> <li>◆ jtest! Command Line Mode</li> <li>◆ jtest! RuleWizard</li> <li>◆ jtest! Security</li> <li>◆ jtest! Test Case Parameterization</li> <li>◆ jtest! Tracer</li> <li>◆ jtest! Unit Testing</li> </ul>
---	--

Linking upgrade license ページの右側にアップグレード ライセンスが表示され、左側にアップグレード ライセンスが参照するメイン ライセンスが表示されます。アップグレード ライセンスとメイン ライセンスの照合は、ツールのバージョンと機能によって自動的に行われます。



2. [Select] リンクをクリックして、2 つのライセンスをリンクします。

**Linking upgrade license**  
[\[Back\]](#) [\[Refresh page\]](#)

**Select the license from the following list:**

**Tool:**           jtest! 7.0    [\[Select\]](#)

**Machine ID:**   UNI-ab85616

**Expiration date:** Unlimited

**Tokens:**        3

**License features:**

- ◆ jtest!
- ◆ jtest! Coding Standards
- ◆ jtest! Command Line Mode
- ◆ jtest! RuleWizard
- ◆ jtest! Security
- ◆ jtest! Test case parametrization
- ◆ jtest! Test case sniffer
- ◆ jtest! Unit Testing

**Upgrade license:**

**Tool:**           jtest! 8.0    [\[Clear\]](#)

**Machine ID:**   UNI-ab85616

**Expiration date:** Unlimited

**Tokens:**        3

**License features:**

- ◆ jtest!
- ◆ jtest! BugDetective
- ◆ jtest! Coding Standards
- ◆ jtest! Command Line Mode
- ◆ jtest! RuleWizard
- ◆ jtest! Security
- ◆ jtest! Test Case Parameterization
- ◆ jtest! Tracer
- ◆ jtest! Unit Testing

[Select] リンクが [Selected] に変わり、メイン ライセンスがアップグレード ライセンスにリンクされたことが分かります。

**Linking upgrade license**  
[\[Back\]](#) [\[Refresh page\]](#)

**Select the license from the following list:**

**Tool:**           jtest! 7.0    **Selected**

**Machine ID:**   UNI-ab85616

**Expiration date:** Unlimited

**Tokens:**        3

**License features:**

- ◆ jtest!
- ◆ jtest! Coding Standards
- ◆ jtest! Command Line Mode
- ◆ jtest! RuleWizard
- ◆ jtest! Security
- ◆ jtest! Test case parametrization
- ◆ jtest! Test case sniffer
- ◆ jtest! Unit Testing

**Upgrade license:**

**Tool:**           jtest! 8.0    [\[Clear\]](#)

**Machine ID:**   UNI-ab85616

**Expiration date:** Unlimited

**Tokens:**        3

**License features:**

- ◆ jtest!
- ◆ jtest! BugDetective
- ◆ jtest! Coding Standards
- ◆ jtest! Command Line Mode
- ◆ jtest! RuleWizard
- ◆ jtest! Security
- ◆ jtest! Test Case Parameterization
- ◆ jtest! Tracer
- ◆ jtest! Unit Testing

[Clear] リンクをクリックすると、アップグレード ライセンスとのリンクを解除することができます。リンクを解除することによって、アップグレード版のツールへの承認されたアクセスが解除されます。

アップグレード ライセンスとメイン ライセンスとのリンクが完了すると、[Not linked upgrade licenses] にあったアップグレードライセンス (この例では Jtest 8.0) が、[Installed Tool Licenses List] のメイン ライセンス (この例では Jtest 7.0) の下に表示されます。

Installed Tool Licenses List					
<a href="#">[Back]</a>					
	Tool name	Machine ID	Expiration date	Available tokens	
1	<a href="#">C++Test 6.0</a>	LINUX2-ab85616	Unlimited	Unlimited	<a href="#">[Remove]</a>
2	<a href="#">C++Test 7.0</a>	UNI-ab85616	Unlimited	1	<a href="#">[Remove]</a>
3	<a href="#">jtest! 7.0</a>	UNI-ab85616	Unlimited	3	<a href="#">[Remove]</a>
	<a href="#">- jtest! 8.0</a>	UNI-ab85616	Unlimited	3	<a href="#">[Remove]</a>
4	<a href="#">jtest! 8.0</a>	UNI-ab85616	03 Oct 2007	1	<a href="#">[Remove]</a>
5	<a href="#">jtest! 8.0</a>	UNI-ab85616	03 Oct 2007	1	<a href="#">[Remove]</a>

## ライセンスについての詳細情報を参照するには

- メイン ライセンスの名前 (この例では jtest! 7.0) をクリックします。ライセンスの詳細が表示されます。

### License details

[\[Back\]](#) [\[Refresh page\]](#)

**Tool:** jtest! 7.0  
**Machine ID:** UNI-4d68b711  
**Expiration date:** Unlimited  
**Tokens (used/available/all):** 0/3/3

#### License features:

- jtest!
- jtest! Coding Standards
- jtest! Command Line Mode
- jtest! RuleWizard
- jtest! Security
- jtest! Test case sniffer
- jtest! Unit Testing

#### Linked licenses:

- [jtest! 8.0 license](#)

#### Filter requests for:

User name:  Host name:  [\[Deny\]](#) [\[Allow\]](#)

OR

Group:  [\[Deny\]](#) [\[Allow\]](#) [\[Edit Groups\]](#)

#### Reserve licenses:

User name:  Host name:  Licenses count:  [\[Apply\]](#)

## ライセンス アクセスのフィルタリング

ライセンス アクセス フィルターを設定するには、各ツールから送信されるのと同じ形式でユーザー名とホスト名を入力します。プラットフォームの違いによって、ツールは異なる形式を使用します。たとえば、1 台のマシンが以下の 3 とおりの名前前で定義される場合があります。

- 10.9.1.50
- gina
- gina.parasoft.com.pl

フィルタリングの際に問題を避けるため、以下のレポートでユーザー / ホスト名の形式を確認することを推奨します。

- トークンの使用状況の確認
- ライセンスの詳細情報の確認

#### 特定のユーザーのライセンス アクセスをフィルタリングするには：

1. [License Details] ページの [Filter requests for] セクションで次の情報を入力します。
  - ユーザー名
  - ホスト名
2. [Deny] または [Allow] ボタンをクリックします。

#### 特定のグループのライセンス アクセスをフィルタリングするには：

1. [License Details] ページの [Filter requests for] セクションでグループ名を選択します。
2. [Deny] または [Allow] ボタンをクリックします。

**注意：**グループの作成と編集については 309 ページの「Pattern Group Management」を参照してください。

ユーザー名またはグループ名が [Filter requests for] セクションに表示されます。必要に応じて 1 つまたはすべてのフィルターを削除することができます。フィルターのアクションは累積されますが、[Deny] ボタンをクリックすると [Allow] のアクションが削除されます。

---

## ライセンスの予約

ライセンスの予約を設定するには、各ツールから送信されるのと同じ形式でユーザー名とホスト名を入力します。プラットフォームの違いによって、ツールは異なる形式を使用します。たとえば、1 台のマシンが以下の 3 とおりの名前前で定義される場合があります。

- 10.9.1.50
- gina
- gina.parasoft.com.pl

予約の際に問題を避けるため、以下のレポートでユーザー / ホスト名の形式を確認することを推奨します。

- 312 ページの「トークンの使用状況の確認」
- 313 ページの「ライセンスの詳細情報の確認」

#### ライセンスを特定ユーザー用に予約するには：

1. [License Details] ページの [Reserve Licenses] セクションで以下の情報を入力します。

- ユーザー名
  - ホスト名
2. 必要なライセンス数を指定します。
  3. [Add] をクリックします。

License Reserved セクションにユーザー名が表示されます。必要に応じて、選択した予約またはすべての予約を削除できます。

## ノードロック ライセンスの管理

マシン ロック ライセンスは、ライセンスの予約の一部です。初めてライセンスを使用する際に、ライセンス請求者の IP に対して完全にライセンス（すべての利用可能なトークン）を予約してください。ホスト名のフィールドでマシン IP を追加して、手動でライセンスを予約することができます。他のマシンによってライセンスが完全に予約されている場合、既存の予約を削除しなければならないことがあります。

---

## Pattern Group Management

新しいパターン グループを追加したり、既存のパターン グループを編集するには、License Server ホーム ページで [Management] > [Manage Pattern Groups] を選択するか、[License Details] ページの [Filter requests for] または [Reserve licenses] セクションの下の [Edit Groups] リンクをクリックします。

[Pattern groups] ページが表示されます。

**Pattern groups**

[\[Back\]](#)

**New pattern group:**  [\[Add\]](#)

**Existing groups:**

- any (p:1, g:0) [\[Edit\]](#) [\[Remove\]](#)

新しい（空の）グループを作成するには、次の操作を行います。

1. 新しいパターン グループを入力し、[Add] をクリックします。

[Existing groups] セクションにグループ名が表示され、グループ名の後の括弧内に次のコードが表示されます。

- **p:** 指定のユーザー名とホストを含むライセンスの数。
- **g:** 指定のユーザー名とホストに対するサブグループとパターンを含むライセンスの数。

既存のグループを編集するには、編集するグループ名の隣の [Edit] ボタンをクリックします。

[Edit pattern group] ページが表示されます。

**Edit pattern group: group1**  
[\[Back\]](#)

**New pattern:** User name:  Host name:  [\[Add\]](#)

**Import:** Patterns (csv):  [Browse...](#) [\[Import\]](#)

**Add group:** Group name:  [\[Add\]](#)

**Existing patterns:**

- mali on host cave [\[Remove\]](#)

**Existing groups:**

- Pattern group grs [\[Remove\]](#)
- +pattern janusz on host .\*

グループにユーザーを追加するには、次の操作を行います。

- 適切なユーザー名とホスト名を入力し、[Add] をクリックします。

グループにユーザーをインポートするには、次の操作を行います。

- [Browse] ボタンを使って CSV ファイルを選択し、[Import] をクリックします。CSV ファイルは、ヘッダーなしで、ユーザー名とホスト名をカンマで区切った書式にします。  
 例：  
 user1,.\*  
 user2,192.168.1.101

既存のサブグループを追加するには、次の操作を行います。

- 適切なグループ名を選択し、[Add] をクリックします。

追加されたすべてのパターンとサブグループが [Existing patterns] および [Existing groups] セクションに表示されます。

---

## ライセンスの削除

License Server の利用可能なライセンスの一覧からライセンスを削除するには、次の操作を行います。

1. Installed Tool Licenses List ページで、削除するツール ライセンスの [Remove] リンクをクリックします。  
 確認のポップアップ ウィンドウが表示されます。
2. [OK] をクリックします。  
 削除したライセンスについての情報が表示されます。

---

## ライセンス要求のリダイレクト

DTP Engine for C/C++ などの Parasoft DTP Engine から出されたライセンス要求を、DTP の別のインスタンスにリダイレクトすることができます。ライセンス要求のリダイレクトは、DTP の 2 つのライセンス間で、あるいは DTP と License Server のスタンドアロン インスタンスの間で行うことができます。

DTP のライセンス要求をリダイレクトする例を挙げます。開発部門内に 2 つの DTP Server があり、それぞれ違う開発チームが使用しています。そして、部門全体で使用するすべての DTP Engine を共通の License Server で管理したいとします。この場合、一方の DTP Server でライセンスのリダイレクトを設定して、もう一方の License Server に要求をリダイレクトすることができます。

1. ライセンス要求をリダイレクトする DTP インスタンスの [DTP\_INSTALL]/conf ディレクトリにある PSTRootConfig.xml 構成ファイルを編集します。

<root-config> タグに以下の行を追加します。http://host:port は、共通の License Server を指します。

```
<license-requests-delegate-url>http://host:port</license-requests-delegate-url>
```

2. PSTRootConfig.xml 構成ファイルを保存し、DTP Server (DTP Server のサービスと DataCollector のサービス) を再起動します。

License Server が実行するデフォルトのポート番号は 2002 です。http:// プロトコル接頭辞を含む完全な URL を必ず使用してください。

---

## 全機能の DTP ライセンスへのアップグレード

ライセンスによって、DTP で利用できる機能が異なる場合があります。全機能を利用できるライセンスにアップグレードする場合、そのライセンスで利用可能になった機能を使用するには、DTP を Server を再起動する必要があります。『Development Testing Platform クイック インストール ガイド』の「Development Testing Platform サービスの開始」を参照してください。このマニュアルは [DTP\_INSTALL]/doc にあります。

---

## ライセンス サマリ

ライセンスの統計を表示するには、License Server ホーム ページで [Management] > [Licenses Summary] を選択します。以下の 2 つのセクションから構成された License Statistics ページが表示されます。

- Tokens usage details: 「トークンの使用状況の確認」を参照。
- Licenses usage details: 「ライセンスの詳細情報の確認」を参照。

License Statistics ページには、ライセンス要求についての統計、トークンの使用状況についての詳細、およびツールごとのトークンの使用数が表示されます。

---

### トークンの使用状況の確認

[Tokens usage details] セクションには、使用されたトークン（ライセンスまたはライセンス機能）の数についての情報が表示されます。たとえば、Jtest は Coding Standards や Unit Testing などのトークンを持ちます。このセクションではトークンのステータスも参照できます。

#### Licenses Summary ページ - Tokens usage details

<p><b>Licenses Summary</b> <a href="#">[Back]</a> <a href="#">[Refresh page]</a></p> <p><b>Tokens usage details:</b></p> <ul style="list-style-type: none"><li>• <a href="#">[View]</a> <b>jtest! 8.0</b> used by <b>jeehongm</b> on host <b>trout</b> 10.10.32.46 - expires in 9 minutes.</li><li>• <a href="#">[View]</a> <b>jtest! 8.0</b> used by <b>nyanm</b> on host <b>holstein</b> 10.10.32.14 - expires in 6 minutes.</li></ul>
--

特定のツールの詳細なトークン使用状況を表示するには、次の手順に従います。

- [\[View\]](#) をクリックします。  
Tokens details ページが表示されます。

## Token details ページ

**Token details**  
[\[Back\]](#) [\[Refresh page\]](#)

**Used by:** jeehongm on host trout (WIN32-307e3eca)  
**Expire in:** less than minute  
**License:** [jtest! 8.0](#) (UNI-598917d5)

**Features:**

- ◆ [jtest!](#)
- ◆ [jtest! BugDetective](#)
- ◆ [jtest! Coding Standards](#)
- ◆ [jtest! Command Line Mode](#)
- ◆ [jtest! RuleWizard](#)
- ◆ [jtest! Security](#)
- ◆ [jtest! Test Case Parameterization](#)  
jtest! Test Case Parametrization (compatible)
- ◆ [jtest! Unit Testing](#)

License Server ver. 2.7 build (02/Oct/2007)

Token details ページにはすべての利用可能なライセンス機能が表示されます。現在使用されている機能は緑色でハイライト表示されます。

---

## ライセンスの詳細情報の確認

[Licenses usage details] セクションには、利用可能なライセンスの License details ページへのリンクがあるほか、ツールごとのトークンの使用状況が表示されます。

## Licenses usage details ページ

**Licenses usage details:**

[BPELMaestro 2.1 For WIN32](#)

- ◆ 0 token s used. Unlimited licenses

[BPELMaestro 2.2 For WIN32](#)

- ◆ 0 token s used. Unlimited licenses

[BPELMaestro 3.0 For UNI](#)

- ◆ 0 tokens used out of 100

[jcontract 1.5 For UNI](#)

- ◆ 0 tokens used out of 40

[jtest! 7.0 For UNI](#) - 0 tokens used out of 50  
[- jtest! 8.0 For UNI](#) - 19 tokens used out of 50

- ◆ 19 tokens used out of 50
- ◆ 30 reserved out of 50

トークンの使用状況の情報は、利用可能なトークンの割合に基づいてカテゴリ分けされています。利用可能なトークンの割合は次のように色で表されます。



- 青：利用可能なトークンが無制限にある。
- 緑：20% から 100% のトークンが利用可能。
- 黄：0% から 20% のトークンが利用可能。
- 赤：現行ライセンスに残っているトークンはない。
- 黒：予約されたトークン。

ライセンスがアップグレードされる時、トークンは新しいバージョンのツールか前バージョンのツールのいずれかに適用されます。しかし、どの時点においても使用できるライセンス数は一定です。たとえば、C++test を 5 ライセンス購入しており、バージョン 7.0 から 8.0 にアップグレードしたとします。アップグレード後も License Server は 5 ライセンスだけを許可します。ユーザーはどのバージョンを使用するかを決定できます。たとえば、3 つのトークンを C++test バージョン 8.0 で使用して残りの 2 つをバージョン 7.0 で使用することも、5 つのすべてのトークンをバージョン 8.0 で使用することもできます。

特定のライセンスのトークンの使用状況を参照するには、そのライセンスのリンクをクリックします。たとえば Jtest 8.0 For UNI のリンクをクリックすると、License details/License usage ページが表示されます。

## License details / License Usage ページ

**License details**  
[\[Back\]](#) [\[Refresh page\]](#)

**Tool:** jtest! 8.0 (Upgrade license)  
**Machine ID:** UNI-4d68b711  
**Expiration date:** 18 Dec 2007  
**Tokens (used/available/all):** 19/31/50 \*

**License features:**

- ◆ jtest!
- ◆ jtest! BugDetective
- ◆ jtest! Coding Standards
- ◆ jtest! Command Line Mode
- ◆ jtest! RuleWizard
- ◆ jtest! Security
- ◆ jtest! Test Case Parameterization
- ◆ jtest! Tracer
  - ◆ jtest! Test case sniffer (compatible)
- ◆ jtest! Unit Testing

License is linked to [jtest! 7.0 license](#). [\[Link to\]](#)  
 \* Reservation and complete usage is available at [main license details page](#).

**License usage:**

- Used by **gmarzec** on host **misio.parasoft.com.pl** 10.9.1.57 (LINUX2) - expires in 14 minutes
- Used by **januszst** on host **argentina** 10.9.1.115 (WIN32) - expires in 11 minutes
- Used by **kerek** on host **elrond** 10.9.1.111 (WIN32) - expires in 8 minutes
- Used by **ksiazek** on host **nazgul** 10.9.1.145 (WIN32) - expires in 6 minutes
- Used by **kuba** on host **zeus** 10.9.1.151 (LINUX2) - expires in 14 minutes
- Used by **maciekg** on host **zero** 10.9.1.68 (WIN32) - expires in 2 minutes
- Used by **michal** on host **Mystery2** 10.9.1.87 (WIN32) - expires in 13 minutes
- Used by **mirek** on host **macleod** 10.9.1.109 (WIN32) - expires in 7 minutes
- Used by **mkuchar** on host **grstest** 10.9.1.93 (WIN32) - expires in 1 minutes
- Used by **nightly** on host **pam.parasoft.com.pl** 10.9.1.38 (LINUX2) - expires in 3 minutes
- Used by **pkoziol** on host **seal** 10.9.1.75 (LINUX2) - expires in 11 minutes
- Used by **ramil** on host **caprica** 10.9.1.33 (WIN32) - expires in 5 minutes
- Used by **szmyru** on host **gobos** 10.9.1.71 (WIN32) - expires in 3 minutes
- 30 reserved for group **all**
  - Used by **mlyko** on host **spock** 10.9.1.105 (WIN32) - expires in 13 minutes
  - Used by **mmartys** on host **ink.parasoft.com.pl** 10.9.1.59 (LINUX2) - expires in 6 minutes
  - Used by **maco** on host **phalanx** 10.9.1.107 (WIN32) - expires in 7 minutes
  - Used by **leon** on host **zero** 10.9.1.68 (WIN32) - expires in 14 minutes

有効期限および使用されているトークンの詳細な情報を見ることができます。トークンの数は次のように色で表されます。

- **黒**：トークンは有効であり、使用されて必要とされる限り、許可されます。
- **紫**：トークンは現在使用中です。しかし、今後ツールがライセンスを要求した場合、トークンは再び確認されます。

---

## サポートされているツールの表示

License Server ページで [View Supported Tools] アイコンをクリックするか、[Management] メニューの [View Supported Tools] をクリックすると、サポートされているツールの一覧が表示されます。インストールした Parasoft 製品がこのバージョンの License Server でサポートされているかどうかを確認できます。サポートされていない場合は、データベースを更新します。

[\[Tools Database\] ページ](#)

### Tools database

[\[Back\]](#)

[\[Update Tools Database\]](#)

- .TEST 1.0, 1.5, 1.6, 2.0, 3.0, 3.1, 4.0, 4.1, 5.0, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 10.0
- BPELMaestro 1.0, 1.5, 1.6, 2.0, 2.1, 2.2, 3.0, 3.1, 3.2, 4.0, 4.1, 5.0, 5.2
- C++Test  
1.2, 1.3, 2.0, 2.1, 2.2, 2.3, 6.0, 6.5, 6.6, 6.7, 7.0, 7.1, 7.2, 7.3, 9.0, 9.1, 9.2, 9.3, 9.4, 9.5, 10.0
- CodeWizard 4.0, 4.1, 4.2, 4.3, 5.0
- DataRecon 1.0, 1.2, 1.3
- Enterprise Float 1.0, 2.0
- Enterprise User 1.0, 2.0
- Environment Manager 2.0, 2.1, 2.2, 2.3, 2.4, 2.5
- GUITest 1.0
- Global Reporting System 1.0, 1.1, 1.2, 2.0
- Insure++ 5.1, 5.2, 6.0, 6.1, 7.0, 7.1, 7.2, 7.3, 7.4
- JContract 1.0, 1.1, 1.2, 1.5
- JTest 4.2, 4.5, 5.0, 6.0, 7.0, 8.0, 9.0, 9.1, 9.2, 9.4, 9.5, 10.0
- LicenseServer 1.0, 2.0
- Load Test 6.0, 9.0, 9.1, 9.2, 9.4, 9.5, 9.6, 9.7

---

## Tool Database の更新

Tool database を更新するには、次の手順に従います。

- [\[Update Tools Database\]](#) をクリックします。

[Tools database update] ページが開き、Web またはファイルからデータベースを更新できます。更新ファイルを電子メールで受け取った場合などは、ファイルから更新できます。

[\[Tools database update\] ページ](#)

**Tools database update**

[\[Back\]](#)

Select update type:

Update from remote server (<http://www.parasoft.com/jsp/templates/misc/licserv2/licserv.enc>)

Update from file:

[\[Update\]](#)

License Server ver. 2.1 build (31/Mar/2006)

## アクセス ログの表示

License Server ホーム ページで [Management] メニューの [View Access Logs] を選択すると、Report Center に送信されたログのリストが表示されます。

### Log Files List ページ

Log files list	
<a href="#">[Back]</a>	
File name	File size [kB]
<a href="#">[View]</a> ls_access.log	259

[View] をクリックすると、次の図のような詳細レポートが開きます。

### Log File ページ

Log file: ls_access.log	
<a href="#">[Back]</a>	
<input type="checkbox"/> Use filter	<input type="text"/> (regexp pattern)
<a href="#">[Refresh view]</a>	
<a href="#">[Bottom]</a>	
10.10.32.35 [2006-03-31 04:50:01.552] AUTHORIZE, n1=0, 0 (OK)	
10.10.36.208 [2006-03-31 04:50:01.552] AUTHORIZE, n1=0, 0 (OK)	
10.10.32.37 [2006-03-31 04:50:01.552] AUTHORIZE, n1=0, 0 (OK)	
10.10.225.75 [2006-03-31 04:50:01.568] AUTHORIZE, n1=0, 0 (OK)	
10.10.32.50 [2006-03-31 04:50:01.552] AUTHORIZE, n1=0, 0 (OK)	
10.10.225.33 [2006-03-31 04:50:01.552] AUTHORIZE, n1=0, 0 (OK)	
10.10.32.34 [2006-03-31 04:50:01.552] AUTHORIZE, n1=0, 0 (OK)	
10.10.36.16 [2006-03-31 04:50:01.568] AUTHORIZE, n1=0, 0 (OK)	
10.10.225.84 [2006-03-31 04:50:01.568] AUTHORIZE, n1=0, 0 (OK)	
10.10.32.37 [2006-03-31 04:54:43.927] AUTHORIZE, n1=0, 0 (OK)	
10.10.225.75 [2006-03-31 04:54:43.927] AUTHORIZE, n1=0, 0 (OK)	
10.10.32.34 [2006-03-31 04:54:43.927] AUTHORIZE, n1=0, 0 (OK)	
10.10.32.35 [2006-03-31 04:54:43.927] AUTHORIZE, n1=0, 0 (OK)	

データをフィルタリングするには、次の操作を行います。

1. Log File ページのログ リストの上にある [Use filter] チェックボックスをオンにします。
2. たとえば「10.10.2」のように正規表現を入力し、Enter キーを押すか、[Refresh] リンクをクリックします。

次の図のようにリストがフィルタリングされます。

## Log File - フィルタリングされたリスト

**Log file: ls\_access.log**

[\[Back\]](#)

Use filter  (regexp pattern)

[\[Refresh view\]](#)

[\[Bottom\]](#)

```
10.10.225.75 [2006-04-04 12:42:12.373] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 12:42:12.373] AUTHORIZE, n1=0, 0 (OK)
10.10.225.75 [2006-04-04 12:45:44.576] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 12:45:44.576] AUTHORIZE, n1=0, 0 (OK)
10.10.225.75 [2006-04-04 02:45:44.560] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 02:45:44.560] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 04:45:44.592] AUTHORIZE, n1=0, 0 (OK)
10.10.225.75 [2006-04-04 04:45:44.607] AUTHORIZE, n1=0, 0 (OK)
10.10.225.75 [2006-04-04 06:45:44.607] AUTHORIZE, n1=0, 0 (OK)
10.10.28.10 [2006-04-04 06:45:44.607] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 06:45:44.654] AUTHORIZE, n1=0, 0 (OK)
10.10.28.10 [2006-04-04 08:45:44.654] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 08:45:44.670] AUTHORIZE, n1=0, 0 (OK)
10.10.225.20 [2006-04-04 08:45:44.670] AUTHORIZE, n1=0, 0 (OK)
10.10.225.75 [2006-04-04 08:45:44.670] AUTHORIZE, n1=0, 0 (OK)
10.10.28.10 [2006-04-04 10:45:44.670] AUTHORIZE, n1=0, 0 (OK)
10.10.225.20 [2006-04-04 10:45:44.685] AUTHORIZE, n1=0, 0 (OK)
10.10.225.33 [2006-04-04 10:45:44.685] AUTHORIZE, n1=0, 0 (OK)
10.10.225.75 [2006-04-04 10:45:44.685] AUTHORIZE, n1=0, 0 (OK)
```

# その他の構成

このセクションの内容：

- LDAP の設定
- シングル サインオンの有効化
- ログイン / ログアウト情報の保存
- SSL を使用した Development Testing Platform の実行
- リバース プロキシのサポート
- .log ファイル プロパティの変更
- マルチキャスト DNS の使用
- 制限モードでの権限とステータス遷移の設定
- Microsoft Word へのエクスポートのカスタマイズ
- 信頼された証明書の追加
- ツールバーでのアプリケーションの有効化と無効化

---

## 制限モードでの権限とステータス遷移の設定

Development Testing Platform (DTP) サーバーは「制限モード」で動作するように構成できます。制限モードでは、プロジェクト成果物に対して実行できるアクションがより厳しくコントロールされます。たとえば：

- ユーザーに表示されるプロジェクト成果物（要求、シナリオ、欠陥、テストセッション）のステータスは、ユーザーの権限とその成果物の現在のステータスによって異なります。たとえば、制限モードではない場合、欠陥のステータスとして New、Assigned、Resolved、および Verified が表示されるでしょう。しかし、制限モードでは、一般のチームメンバーに権限を設定して、彼らに変更できるステータスを「New から Assigned へ」または「Assigned から Resolved へ」だけにすることができます。そうすれば、一般メンバーは成果物のステータスを "Verified" に設定したり、"New" から直接 "Resolved" にステータスを変更したりできません。
- ユーザーはプロジェクトの反復を削除することはできませんが、回避策として、不要な成果物を特別に名付けられたプロジェクト（たとえば "Obsolete" プロジェクト）に移動することができます。適切な権限のあるユーザーはシナリオとシナリオセットを削除できます。
- プロジェクトは削除できません。プロジェクトを削除するのではなく、プロジェクトを編集して非アクティブにします。非アクティブのプロジェクトは、Report Center と Project Center のプロジェクトリストに表示されません。

---

## 制限モードの有効化

1. 管理者権限を使って、DTP Server マシン上の DTP\_HOME\conf\PSTRootConfig.xml ファイルを編集します。
2. <root-config> の <strict-mode> を false から true に変更します。  
<strict-mode>true</strict-mode>
3. DTP Server を再起動します。

---

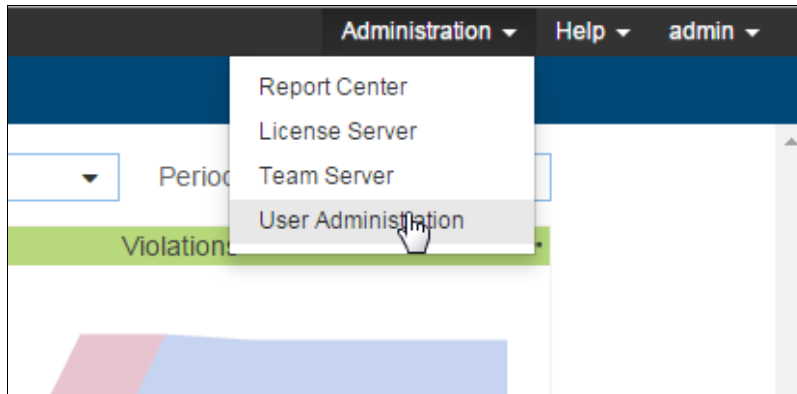
## プロジェクト成果物のステータス遷移のための権限追加

DTP を制限モードに切り替えた後は、特別な権限を持つユーザーだけがプロジェクト成果物のステータスを変更できます。ステータス遷移の権限が与えられない場合、ユーザーはどのステータスも変更できません。

特定のユーザーまたはチーム（ユーザーグループ）にステータス遷移の権限を追加するには、次の操作を行います。



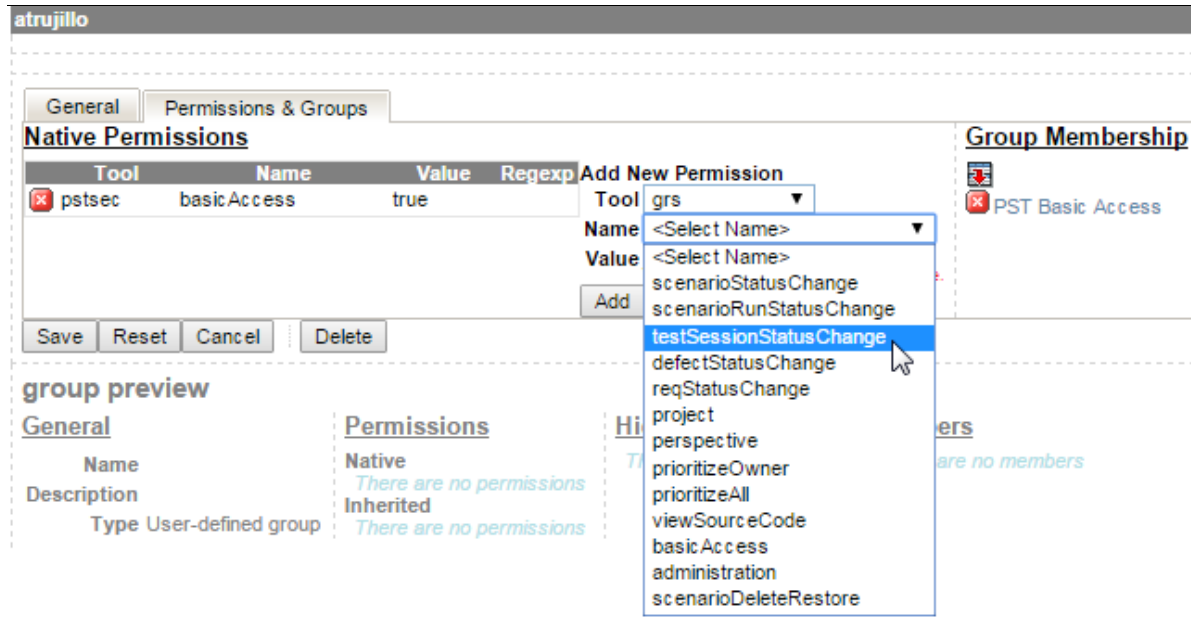
1. [ 管理 ] ドロップダウン メニューから [User Administration] を選択して User Administration モジュールに移動します。管理者権限が必要です。



2. [ ユーザー ] タブをクリックし、編集したいユーザーの [ 編集 ] ボタンをクリックします。  
グループを作成してユーザーを追加することで、一度に複数のユーザーを変更できます。一般的な User Administration の使用方法と権限については 249 ページの「[User Administration] ページ」で説明しています。
  - 一人のユーザーが複数のグループに属することができます。
  - デフォルトで "Sample Strict Mode Permissions" というグループが用意されています。このグループはすべてのステータス遷移権限を所有します。特定のユーザーにすべてのステータス遷移権限を与える最も簡単な方法は、そのユーザーをこのグループに追加することです。このグループにはデフォルトでユーザー "admin" が追加されます。つまり、admin はすべてのステータス遷移権限をデフォルトで所有します。
3. 次の操作を行って、特定のステータス遷移権限だけを許可します。
  - a. 特定のユーザーまたはグループの編集画面で [ 権限とグループ ] タブをクリックします。[ ツール ] > [grs] のための新しい権限を追加します。[ ツール ] > [grs] を選択すると、以下のステータス遷移権限を利用できます。

defectStatusChange ( 欠陥 用 )  
 reqStatusChange ( 要求 用 )  
 scenarioStatusChange ( シナリオ 用 )

testSessionStatusChange ( テスト セッション 用 )



- b. 設定するステータス遷移権限を選択します。たとえばテストセッションのための権限を設定するのであれば testSessionStatusChange を選択します。そして、変更するステータス権限を設定します。たとえば、この権限を持つユーザーがアイテムの権限を "Draft" から "Read for Review" に変更できるようにする場合、[Draft -> Ready for Review] を選択します。利用できるステータス権限は、チームに対して設定したステータスによって異なります。

たとえば、シナリオのステータス権限を指定するものとします。この場合、scenarioStatusChange を選択します。[ 値 ] のドロップダウン リストに、利用できるシナリオ ステータス遷移の値がすべて表示されます。

新規権限の追加

ツール grs

名前 scenarioStatusChange

値 [unspecified]->Draft  regexp

Predefined values

[unspecified]->Draft

追加

- [unspecified]->Draft
- [unspecified]->Ready for Review
- [unspecified]->Ready for QA
- [unspecified]->Approved
- Draft->Ready for Review
- Draft->Ready for QA
- Draft->Approved
- Ready for Review->Draft
- Ready for Review->Ready for QA
- Ready for Review->Approved
- Ready for QA->Draft
- Ready for QA->Ready for Review
- Ready for QA->Approved
- Approved->Draft
- Approved->Ready for Review
- Approved->Ready for QA

- ユーザーがシナリオを作成できるようにするには、scenarioStatusChange を選択して何らかの "[unspecified]->" 権限を許可する必要があります。たとえば "[unspecified]-> Draft" の場合、このユーザーには新しいシナリオの作成ページで Draft ステータスだけが表示されます。どの "[unspecified]->" 権限も許可しない場合、このユーザーは指定された種類のエンティティを新規に作成できません。
- "Draft -> Ready for Review" および "Draft -> Ready for QA" をユーザーに許可した場合、このユーザーは Draft ステータスのシナリオを編集するときに、"Ready for Review" または "Ready for QA" にだけステータスを変更できます。

---

# Microsoft Word へのエクスポートのカスタマイズ

一部のレポートには、Microsoft Word 文書 (docx) にエクスポートするオプションがあります。[Export to Word] ボタンをクリックすると、現在のレポートが docx ファイルに開きます。この docx ファイルには、Development Testing Platform (DTP) のヘッダーとフッターが定義された Microsoft Word テンプレートが使用されます。

たとえば会社独自のテンプレートなど、カスタム テンプレートを使用する場合は次の操作を行います。

1. カスタム テンプレート ファイル (.docx) を `$DTP_HOME/grs/xreports/planning/common/docx` ディレクトリにコピーします。
2. ファイル名を `template.docx` に変更します。

---

## 信頼された証明書の追加

Development Testing Platform (DTP) を外部のツールやサービスと統合するには、TLS/SSL でセキュリティ保護された接続が必要な場合があります。サーバーの証明書が信頼されない場合、または信頼された認証局によって署名されていない場合、DTP は外部サーバーへの接続を拒否することがあります。新しい信頼された証明書を追加するには、次の操作を行います。

1. 信頼された証明書を取得します。この証明書の形式は、Java keytool アプリケーションが承認する任意の形式で構いません。
2. 次のコマンドを実行してトラストストアに証明書をインポートします。

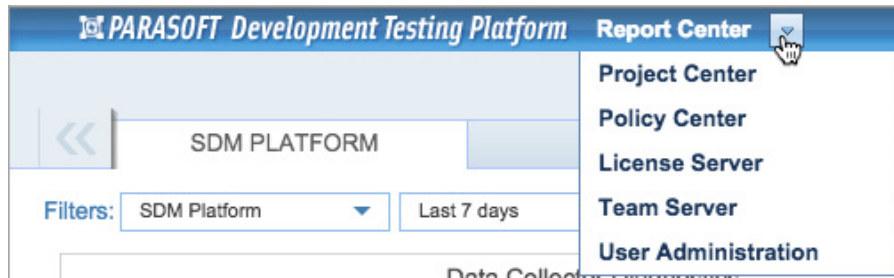
```
keytool -import -alias <new unique alias> -file <certificate file> -keystore  
<DTP_INSTALL>/jre/lib/security/cacerts
```

3. パスワードを求められたら "changeit" と入力します。
4. 証明書のインポートの確認を求められたら、インポートを承認します。
5. DTP サービスを再開して変更を適用します。

証明書チェーンをインポートするには、証明書ごとにステップ 1 から 4 を繰り返します。順番は、1 番目がルート証明書、最後がエンティティ証明書です。

# ツールバーでのアプリケーションの無効化と有効化

Report Center や Project Center といったアプリケーションを DTP のツールバーで無効化することができます。



1. [INSTALLATION\_DIR]\conf\PSTRootConfig.xml 構成ファイルをエディターで開きます。
2. <visible-apps> セクションのコメントを外し、ツールバーで無効化するアプリケーションには false を、有効化するアプリケーションには true を設定します。

<visible-apps> セクションが存在しない場合、<root-config> ノードの下に手動で追加する必要があります。

```
<!-- Configures which web application can be visible in DTP head menu -->
```

```
<visible-apps>
  <report-center>true</report-center>
  <project-center>true</project-center>
  <policy-center>true</policy-center>
  <team-server>true</team-server>
  <license-server>true</license-server>
  <user-administration>true</user-administration>
</visible-apps>
```

3. 編集が完了したら、DTP Server を再起動する必要があります。

各アプリケーションのライセンスも必要です。また、個々のユーザーは有効化されたアプリケーションを見るための権限が必要です ( 249 ページの「[User Administration] ページ」を参照してください)。

# Development Testing Platform の統合

このセクションの内容：

- Parasoft Test から Report Center/Project Center へのテスト結果の送信
- CSV からの BTS データのインポート
- BTS/RMS Scanner と Updater
- Report Center および Project Center とサードパーティ製ツールの統合
- DTP イベントのサブスクリプション

---

# Parasoft Test から Report Center/ Project Center へのテスト結果の送信

Jtest、C++test、dotTEST、SOAtest といった Parasoft 製品からテスト結果をインポートするには、接続パラメーターを適切に設定する必要があります。

1. Parasoft Test 製品で [Parasoft] メニューの [設定] をクリックします。
2. [Parasoft] > [Development Testing Platform] を選択します。
3. 適切な接続プロパティを指定します。詳細については『Parasoft Test ユーザーズ ガイド』の「Parasoft Concerto および Parasoft Concerto Report Center との接続」を参照してください。
4. [適用] をクリックします。
5. [OK] をクリックして設定を保存します。



# CSV からの BTS データのインポート

DTP は、CSV ファイル (カンマ区切りファイル) を使用して任意のバグ トラッキング システムと統合することができます。バグ トラッキング システムが CSV ファイルにエクスポートできる欠陥 / 改善であれば、DTP にインポートできます。

CSV ファイルを使用して DTP と BTS/RMS を統合するには、以下のセクションで説明する手順を実行します。

- CSV ファイルの準備
- CSV ファイルの書式
- BTS Scanner 構成ファイルの準備

**注意:** CSV ファイルが、たとえば欠陥の名前などに national characters を使用している場合、DTP にインポートする前にファイルを UTF-8 エンコーディングで保存してください。

また、.csv、.xls、または .xlsx ファイルから DTP のプロジェクトに内部的な欠陥をインポートすることもできます。ファイルから欠陥をインポートすると、DTP はそれを内部的な欠陥として認識します。その結果、より多くのプロパティが定義できるようになります。詳細については 20 ページの「ファイルからの欠陥のインポート」を参照してください。

## CSV ファイルの準備

CSV を使用して DTP と BTS/RMS を統合するには、次の 2 つのファイルが必要です。

- **CSV\_bugs.csv:** BTS の欠陥のリストを含むファイルです。
- **CSV\_activities.csv:** 欠陥のステータスの変更履歴を含むファイルです。

DTP\grs\extras\csv-bts にファイルのサンプルがあります。

BTS Scanner ジョブは定期的にこの 2 つのファイルを読み込んでデータを DTP に送信します (デフォルトの設定では 20 分ごとです。または DTP の Web インターフェイスで指定された値です)。

DTP の欠陥データが最新の状態に保たれるよう、ファイルを定期的に更新すること (つまり BTS からデータをエクスポートすること) を推奨します。一日一回、あるいは理想的には BTS で変更が行われるたびにファイルを更新することを推奨します。

以下のセクションでは BTS 統合で使用する CSV ファイルの作成方法について説明します。

### CSV 欠陥ファイルの作成

DTP は CSV 欠陥ファイルから欠陥 / 改善の基本的な情報を読み取ります。例として、ファイル名が CSV\_bugs.csv であると仮定します。

A	B	C	D	E	F	G	H	I
ID	Type	Summary	Status	Created On	Owner	Priority	Severity	Resolution
40174	defect	Problem with add-ins.	VERIFIED	12/21/09 06:26 AM	user2	medium	Low	FIXED
40185	enhancement	Reorganize categories in HTML Report	NEW	12/21/09 12:26 PM	user4	high		
40175	defect	Unit testing for Z project fails.	RESOLVED	12/21/09 07:06 AM		medium	High	FIXED
40176	defect	[internal] Update plug-ins	RESOLVED	12/21/09 09:49 AM	user5	medium	Medium	FIXED

DTP がこのファイルを定期的にスキャンする点に注意してください。

どのデータを CSV\_bugs.csv ファイルにエクスポートするかについては、次の 2 種類の方法が考えられます。

- CSV\_bugs.csv ファイルを定期的に削除し、最近変更された欠陥や新しく追加された欠陥の中で、DTP にインポートしたい欠陥だけを保存します。

たとえば、すでに DTP にインポート済みの欠陥のうち 1 つの欠陥のステータスが変更され、同時にまだ DTP にインポートされていない別の欠陥が 1 つ BTS に追加されている場合、CSV\_bugs.csv ファイルには 2 行のデータがあり、この 2 つの欠陥に関する最新の情報が記載されます。

- BTS のすべての欠陥を CSV\_bugs.csv ファイルに書き出します。DTP は定期スキャンを実行する際、最終スキャン以降に変更または追加された欠陥の情報だけを更新します。

**注意：** DTP にインポートされた欠陥は永久に削除されません。CSV\_bugs.csv ファイルを更新することで、欠陥のプロパティの更新と新たな欠陥の追加だけを行うことができます。

CSV\_bugs.csv ファイルの列：

以下のプロパティは必須です。

- **ID** バグ トラッキング システムでの欠陥 / 改善の識別子。
- **Type** 項目が欠陥か改善かを表します。この列の値は、「ix) <bts>/<feature-request>」で説明する <feature-request> タグに指定された項目のいずれかと一致する場合にだけフィーチャーと解釈されます。一致しない場合、値は欠陥と解釈されます。
- **Summary** 欠陥の概要。
- **Status** 欠陥のステータスが open か closed かを表します。「vi) <bts>/<resolved-status>」も参照してください。
- **Created On** BTS で欠陥が作成された日

以下のプロパティは DTP レポートに表示されるため、CSV ファイルに含めることを強く推奨します。

- **Owner** 現在 BTS で欠陥に割り当てられている担当者  
**ヒント：** Owner プロパティには DTP のログイン名か、DTP の Users Administration モジュールに定義された E-mail アドレスと一致するユーザー名を使用することを推奨します。そうすることで、Project Center モジュールで欠陥 / 改善からタスクをスケジュールし、DTP の所有者に割り当てるのが容易になります。
- **Priority/Severity** 欠陥の優先度
- **Resolution** 欠陥が解決された方法。未解決の場合は空にしておきます。「vii) <bts>/<inactive-resolution>」も参照してください。

以下のプロパティ (列) を CSV\_bugs.csv ファイルに含めることができます。

Version、Milestone、Hardware、Os、Project、および Component

これらのプロパティは将来の使用に備えて DTP にインポートされます。次の場所にこれらのプロパティを含むサンプル ファイルがあります。

DTP\grs\extras\csv-bts\CSV\_bugs\_ext.csv

## CSV 欠陥履歴ファイルの作成

DTP は CSV 欠陥履歴ファイルから欠陥 / 改善に加えられたステータス変更の履歴を読み取ります。例として、ファイル名が CSV\_activities.csv であると仮定します。

**注意：** ステータス以外のフィールドの変更をファイルに含めることもできますが、それらの情報は現在のバージョンの DTP では使用されません。

CSV\_bugs.csv ファイルの場合と同じように、どのデータを CSV\_activities.csv ファイルにエクスポートするかについて 2 種類の方法が考えられます。

- CSV\_activities.csv ファイルを定期的に削除し、ファイルには DTP にインポートしたい最近行われたステータス変更だけを保存します。
- すべてのステータス変更履歴を CSV\_activities.csv ファイルに書き出します。DTP は定期スキャンを実行する際、最終スキャン以降に追加された情報だけを更新します。

ステータス変更履歴は Report Center レポートに表示されます。たとえば、Tester Dashboard はプロジェクトの欠陥が時間の経過につれてどのように変化したかという傾向を表すグラフを表示します。

このファイルに `_gresolution_h` プロパティを保存することもできます。このプロパティは特定の欠陥が非アクティブ状態にされたかどうかを指定するのに使われます。

ID	User	Activity Date	Changed Field	Old Value	New Value
40174	user5	01/17/10 06:05 AM	Status	NEW	RESOLVED
40174	user5	01/25/10 01:18 AM	Status	RESOLVED	VERIFIED
40175	user5	01/23/10 07:03 AM	Status	NEW	ASSIGNED
40175	user1	01/23/10 07:44 AM	Status	ASSIGNED	RESOLVED
40176	user5	01/26/10 09:50 AM	Status	NEW	RESOLVED

CSV\_activities.csv ファイルの列：

履歴ファイルには以下の情報を含める必要があります。

- **ID** 欠陥の ID
- **User** 変更を行ったユーザー
- **Activity Date** 変更が行われた日
- **Changed Field** 変更された属性。実際に DTP レポートで使用されるのはステータス属性ですが、所有者や解決方法など他の属性を含めて DTP にインポートすることもできます。
- **Old Value** 変更前のフィールドの値
- **New Value** 変更後のフィールドの値

## CSV ファイルの書式

CSV ファイルには以下の書式ルールを適用します。

- 各値をカンマで区切ります。値にカンマが含まれる場合、引用符で囲みます。
- 引用符はダブルクォーテーションを使用します。
- 行の最後のフィールドの後にはカンマを付けません。

- Null または空フィールドは 2 つのカンマで表します。
- 通常、ファイルの末尾は 1 つの END-OF-LINE 文字で終わります。

## BTS Scanner 構成ファイルの準備

DTP BTS Scanner ジョブの構成ファイルが必要です。

構成ファイルのサンプルが次の場所にあります。

DTP\grs\config\bts\examples\ExampleCSVScannerConfig.xml

ファイルをコピーし、以下で説明するすべてのパラメーターを変更してください。

変更が完了したら、ファイルを DTP\grs\config\bts ディレクトリに置きます。DTP はこのフォルダーを定期的にスキャンし、発見した構成ファイルに基づいて BTS をスキャンします。

BTS Scanner で CSV ファイルをスキャンするには、以下の属性値を指定します。

### i) `<bts>/<name>`

BTS を識別するためにユーザーが見て分かる名前を指定します。

例 :My BTS Server

この名前は Project Center の [ 欠陥 / 改善検索 ] ページに [ 欠陥トラッキング システム ] ドロップダウン リストの選択肢として表示されます。

### ii) `<bts>/<bugs-file>`

CSV\_bugs.csv ファイルの場所

### iii) `<bts>/<activities-file>`

CSV\_activities.csv ファイルの場所

### iv) `<bts>/<fields-mapping>`

このセクションでは CSV\_bugs.csv ファイルの列名を指定します。ID から Created On 列は必須です。Owner から Severity 列を指定することを強く推奨します。残りの列はオプションであり、将来の使用に備えて DTP にインポートされます。

```
<fields-mapping>
  <id>ID</id>
  <bug-type>Type</bug-type>
  <summary>Summary</summary>
  <status>Status</status>
  <creation-date>Created On</creation-date>
  <assigned-to>Owner</assigned-to>
  <priority>Priority</priority>
  <severity>Severity</severity>
  <resolution>Resolution</resolution>

  <version>Version</version>
  <milestone>Milestone</milestone>
  <hardware>Hardware</hardware>
```

```

    <os>Os</os>
    <project>Project</project>
    <component>Component</component>
</fields-mapping>

```

#### v) **<bts>/<activity-fields-mapping>**

このセクションでは CSV\_activities.csv ファイルの列名を指定します。すべての列が CSV\_activities.csv 中に存在する必要があります。

```

<activity-fields-mapping>
  <id>ID</id>
  <who>User</who>
  <date>Activity Date</date>
  <changed-field>Changed Field</changed-field>
  <old-value>Old Value</old-value>
  <new-value>New Value</new-value>
</activity-fields-mapping>

```

#### vi) **<bts>/<resolved-status>**

欠陥が解決されたことを表す値を指定します。つまり、欠陥の修正または改善の実装を表す値を指定します。

この情報は DTP のさまざまなレポートで使用されます。たとえば、Tester Dashboard グラフは、オープン中の欠陥と未テスト / テスト済みの欠陥の割合が時間の経過につれどのように変化したかを表示します。クローズ済みの欠陥が未テスト / テスト済みの欠陥として扱われます。

例：

```

<resolved-status>
  <status>RESOLVED</status>
  <status>CLOSED</status>
</resolved-status>

```

#### vii) **<bts>/<inactive-resolution>**

欠陥が非アクティブであることを表す値を指定します。非アクティブの欠陥は DTP にインポートされません。

例：

```

<inactive-resolution>
  <resolution>DUPLICATE</resolution>
  <resolution>INVALID</resolution>
</inactive-resolution>

```

#### viii) **<bts>/<date-formats>**

CSV\_bugs.csv および CSV\_activities.csv ファイルで使われる日付書式を指定します。最も一般的に使用される書式があらかじめ指定されています。

```

<date-formats>
  <date-format>yyyy-MM-dd hh:mm</date-format>
  <date-format>yyyy-MM-dd hh:mm:ss.0</date-format>
  <date-format>yy MMM-dd hh:mm</date-format>
  <date-format>MM/dd/yy hh:mm aa</date-format>
</date-formats>

```

書式は Java の SimpleDateFormat クラスに定義された書式のいずれかでなければなりません (<http://java.sun.com/javase/6/docs/api/java/text/SimpleDateFormat.html> を参照)。

#### ix) <bts>/<feature-request>

<item> ノードの値は、CSV\_bugs.csv ファイルの \_gType\_h 列の値のうち、欠陥ではなく改善として扱う値を指定します。

例：

```
<feature-request>
  <item>enhancement</item>
</feature-request>
```

#### インポートされた CSV 欠陥 / 改善の検証方法

CSVScannerConfig.xml の変更が完了したら、ファイルを DTP\grs\config\bts ディレクトリに置きます。DTP はこのフォルダーを定期的にスキャンし、見つかった構成ファイルに基づいて BTS をスキャンします。DTP が BTS 構成ファイルをスキャンする方法の詳細については 345 ページの「BTS Scanner の実行」を参照してください。BTS Scanner または RMS Scanner を構成する方法は 2 つあります。

---

# バグ トラッキング システムおよび要求管理システムの統合

Development Testing Platform (DTP) は、BTS Scanner または RMS Scanner を使用して、バグ トラッキング システム (BTS) や要求管理システム (RMS) と統合します。BTS Scanner と RMS Scanner は内部的なコンポーネントであり、DTP と統合する BTS/RMS ごとに構成する必要があります。BTS Scanner と RMS Scanner は、サポートされるシステムからデータベースに、バグと要求のデータをそれぞれ転送します。

各 Scanner は、BTS または RMS のデータベースから重要な情報を取得して、Report Center データベースに送ります。転送されたデータは、バグまたは要求に関連するレポートを作成するために Report Center と Project Center が使用します。構成が完了すると、BTS/RMS Scanner はデフォルトの 15 分間隔で自動的に実行されます。また、必要なときに手動で実行することもできます。

---

## ビルトイン UI を使った BTS/RMS Scanner の構成

Report Center または Project Center の管理ページから BTS Scanner と RMS Scanner を構成することができます。この構成は以下のシステムについてだけ行うことができます。

### バグ トラッキング システム

- Bugzilla
- JIRA 4.x
- JIRA 5.x/6.x (DTP Server で実行するには SSL が必要。詳細については 758 ページの「SSL を使用した Development Testing Platform の実行」を参照)

### 要求管理システム

- Blueprint
- DOORS

---

## BTS 構成の追加

1. 管理ページを開き、[ 設定 ] > [BTS Scanners] を選択します。
2. [ タイプ ] ドロップダウン メニューからシステムを選択します。

3. 新しい構成の名前を入力し、[ 新規作成 ] をクリックします。

The screenshot shows the 'BTS Scanner Configurations' page. At the top, there are navigation tabs for 'Main', 'Projects', and 'Settings'. Below the tabs, the page title is 'BTS Scanner Configurations'. Underneath, it says 'Create a new configuration'. There is a 'Type' dropdown menu set to '--- Bug Tracking Systems ---' and a 'Name' text input field. A 'Create New' button is located below the input fields. Below the form, there is a table with 4 items. The table has columns for 'Modify', 'Migrate', 'Name', 'BTS ID', 'Type', and an unlabeled column. The rows contain configuration details for 'asdf', 'BZ test scanner config', 'Jira 5', and 'test'.

Modify	Migrate	Name	BTS ID	Type	
<a href="#">Modify</a>	<a href="#">Migrate to JIRA 5.x</a>	asdf	31	JIRA 4.x	JIRA-13-07-10_0
<a href="#">Modify</a>		BZ test scanner config	32	Bugzilla	BUGZILLA-13-10
<a href="#">Modify</a>		Jira 5	33	JIRA 5.x	JIRA5-13-04-22_
<a href="#">Modify</a>	<a href="#">Migrate to JIRA 5.x</a>	test	34	JIRA 4.x	JIRA-13-07-10_0

## BTS Scanner の構成

1. BTS Scanner の管理ページを開き、[ 修正 ] 列の [ 修正 ] をクリックします。
2. サーバー設定とログイン情報を入力します。













The screenshot shows the 'Edit BTS Scanner Configuration' page. At the top, there are navigation tabs for 'Main', 'Projects', and 'Settings'. Below the tabs, the page title is 'Edit BTS Scanner Configuration'. Underneath, it says 'JIRA 5.x configuration'. There are two main sections: 'Scanner Setting' and 'Development Testing Platform Setting'. The 'Scanner Setting' section has fields for 'Name' (Jira 5), 'BTS ID' (32), 'JIRA URL' (http://jira2.parasoft.com), 'Username' (devtest), and 'Password' (masked with asterisks). A 'Check' button is below these fields. The 'Development Testing Platform Setting' section has fields for 'DTP Username' (admin) and 'DTP Password' (masked with asterisks). A 'Check' button is below these fields.



3. 成果物のステータスを DTP にマッピングする方法を設定します。

JIRA statuses that are mapped to Concerto

6 Item(s)


Status		
	Closed	
	Open	
	In Progress	
	Reopened	
	Duplicate	
	Won't Fix	

4. 設定が完了したら [ 保存 ] をクリックします。

## RMS 構成の追加

1. 管理ページを開き、[ 設定 ] > [ RMS Scanners ] を選択します。
2. [ タイプ ] ドロップダウン メニューからシステムを選択します。
3. 新しい構成の名前を入力し、[ 新規作成 ] をクリックします。

Main Projects Settings

RMS Scanner Configurations 

Create a new configuration

Type:

Name:


2 Item(s)

Modify	Name	RMS ID	Type	File Name
<a href="#">Modify</a>	Blueprint 2	5	Blueprint	BLUEPRINT-13-05-23_043747.x
<a href="#">Modify</a>	My Blueprint	6	Blueprint	BLUEPRINT-13-06-07_035049.x

## RMS Scanner の構成

1. RMS Scanner の管理ページを開き、[ 修正 ] 列の [ 修正 ] をクリックします。

## 2. サーバー設定とログイン情報を入力します。

 **Edit RMS Scanner Configuration**

Blueprint configuration

Scanner Setting

Name:  ?

RMS ID:

Blueprint URL:  ?

Username:

Password:

Development Testing Platform Setting

DTP Username:  ?

DTP Password:

Import Settings

Target Project:  ?

## UI による IBM Rational DOORS の構成

DOORS の構成ページには、そのシステムに固有のパラメーターが含まれます。

1. 上記の新しい構成を追加する操作手順に従います。
2. RMS Scanner の構成ページを開き、[ 修正 ] 列の [ 修正 ] をクリックします。
3. サーバー設定とログイン情報を入力します。

4. [ 追加 ] ボタンをクリックし、要求を DTP Server にインポートする DOORS モジュールを定義します。指定するパスは、DOORS のフォーマル モジュールまたはリンク モジュールを指す必要があります。たとえば DOORS フォルダーを指してはいけません。

DOORS configuration

General

Name: DOORS rms scanner test

BTS ID: Database Error

DOORS exe: c:\Program Files\IBM\Rational\DOORS\9.2\bin\doors.x

User: Administrator

Password: qaz123

Modules to scan

3 Item(s) Add

Delete	Module name*	DTP project*	DOORS DB Example: 29998@localhost	User	Password
<input checked="" type="checkbox"/>	/Test/Requirements/System Requirements	Default Project			
<input checked="" type="checkbox"/>	/Sports utility vehicle 4x2/Requirements/Functional Requirements	Default Project	29998@localhost	Eric McCall	Training
<input checked="" type="checkbox"/>	/Sports utility vehicle 4x2/Requirements/User Requirements	Default Project	29998@localhost	Eric McCall	Training

5. さらに DOORS の統合情報を指定します。
- **DTP プロジェクト**：指定した DOORS モジュールの要求をインポートする DTP プロジェクト。
  - **DOORS DB**：(任意) 代替 DOORS データベース。
  - **注意**：対応するモジュール名が、DOORS Exe にリンクされている以外の DOORS db で定義されている場合にだけ (General 構成で指定)、このフィールドを使用します。
  - **ユーザー**：DOORS データベースにアクセスでき、Module Name のすべてのオブジェクトに対する読み取り権限を持った有効な DOORS ユーザー名。
  - **パスワード**：User の有効なパスワード。
6. [ 保存 ] をクリックします。

新しい DOORS BTS Scanner 構成ファイルが \$DTP\_HOME/grs/config/bts ディレクトリに作成されます。

## 手動による BTS/RMS Scanner の構成

DTP\_HOME/grs/config/bts ディレクトリに格納された XML ファイルを編集して、BTS Scanner と RMS Scanner を手動で構成することができます。BTS Scanner と RMS Scanner の構成と使用方法については 342 ページの「BTS/RMS Scanner と Updater」を参照してください。

DTP はいくつかのシステムとの単純な統合をサポートします。以下のセクションでは、BTS/RMS Scanner を手動で構成し、各システムと連携して使用方法について説明します。

- HP Quality Center との統合
- Bugzilla との統合

- IBM Rational ClearQuest との統合
- Atlassian JIRA との統合
- IBM Rational Change および Rational Synergy との統合

# BTS/RMS Scanner と Updater

手動で BTS Scanner を構成し、BTS Scanner の .xml 構成ファイルを編集して BTS/RMS を BTS Scanner でスキャンすることができます。この .xml 構成ファイルは、DTP Server がインストールされているホストの `DTP_HOME/grs/config/bts` に格納されています。

この章では次のトピックについて説明します。

- サンプル構成ファイルの準備
- 構成ファイルの追加
- カスタム BTS Scanner/Updater
- BTS Scanner の実行
- BTS Scanner の動作確認
- BTS Updater の構成
- オリジナル ID の使用

## サンプル構成ファイルの準備

`DTP_HOME/grs/config/bts/examples` フォルダには、サポート対象 BTS/RMS ごとにサンプル構成ファイルが用意されています。

サンプル構成ファイルを準備するには、次の操作を行います。

1. サンプル構成ファイルのディレクトリに移動します。  
`DTP_HOME/grs/config/bts/examples`
2. 使用している BTS 用のサンプル ファイルを選択します。  
たとえば、BTS Scanner を JIRA と統合するには、次のファイルを選択します。  
`ExampleJiraScannerConfig.xml`
3. 選択したサンプル ファイルを同じディレクトリにコピーします。  
`DTP_HOME/grs/config/bts/examples`
4. ファイル名を変更します。  
たとえば、`ExampleJiraScannerConfig.xml` を `MyJiraScannerConfig.xml` に変更します。
5. 構成ファイルを編集し、ベンダーに合わせて BTS/RMS 設定を指定します。

BTS Scanner は現時点でいくつかの BTS/RMS をサポートしていますが、それぞれ必要な構成は異なります。

BTS Scanner と特定の BTS/RMS を連携するための構成の詳細については、以下を参照してください。

- HP Quality Center との統合
- Bugzilla との統合
- IBM Rational ClearQuest との統合
- Atlassian JIRA との統合 --older versions

BTS Scanner は CSV ファイルから欠陥をインポートすることもできるので、データを CSV ファイルにエクスポートする機能のある BTS/RMS であれば、DTP と統合できます。BTS Scanner で CSV ファイルをインポートする方法の詳細については、330 ページの「CSV からの BTS データのインポート」を参照してください。

**ヒント:** DTP では統合する成果物の名前をカスタマイズできます。BTS Scanner と BTS/RMS の統合をセットアップした後に、DTP で使う成果物名を BTS/RMS で使われている名前に合わせると便利です。詳細については 274 ページの「成果物の名前変更」を参照してください。

---

## 構成ファイルの追加

BTS/RMS に合わせて構成ファイルを修正したら、BTS Scanner で構成ファイルを読み込みます。BTS Scanner は定期的に行われ、`DTP_HOME/grs/config/bts` にある構成ファイル进行处理します。

新規に DTP をインストールした場合、`bts` ディレクトリは空です。このディレクトリには複数の構成ファイルを追加できます。そのため、BTS Scanner は複数の異なる種類の BTS/RMS に接続することも、同じ種類でパラメーターの異なる複数の BTS/RMS インスタンスに接続することもできます。`bts` ディレクトリに追加された各構成ファイルは、それぞれ 1 つの BTS/RMS インスタンスとの統合に相当します。

準備した構成ファイルを追加して BTS Scanner で読み込むには、次の操作を行います。

- 選択したファイルを `DTP_HOME/grs/config/bts` にコピーまたは移動します。

---

## カスタム BTS Scanner/Updater

次の Java API を実装してカスタム BTS Scanner/Updater を作成することができます。

- `com.parasoft.api.dtp.defects.scan.DefectScanner`
- `com.parasoft.api.dtp.defects.scan.DefectScannerFactory`
- `com.parasoft.api.dtp.defects.update.DefectUpdater`
- `com.parasoft.api.dtp.defects.update.DefectUpdaterFactory`

Java API ドキュメントへのアクセスについては 864 ページの「Java API」を参照してください。ログに関する情報については 416 ページの「ログ ファイルの参照」を参照してください。

---

## カスタム実装のインストール

1. カスタム BTS Scanner/Updater のための jar ファイルを作成します。
2. この jar ファイルを `$DTP_HOME/tomcat/webapps/grs/WEB-INF/lib` ディレクトリにコピーします。
3. DTP を再起動します。

---

## カスタム実装を使用するよう BTS Scanner を構成する

\$DTP\_HOME/grs/config/bts に XML 構成ファイルを作成する必要があります。テンプレートとして \$DTP\_HOME/grs/config/bts/examples/ExampleCustomScannerConfig.xml を使用できます。

カスタム XML 構成ファイルで最も重要なのは次の 2 箇所です。

- scannerFactoryClass 要素の値は、  
com.parasoft.api.dtp.defects.scan.DefectScannerFactory インターフェイスを実装するクラス名に設定しなければなりません。
- updatorFactoryClass 要素の値は、  
com.parasoft.api.dtp.defects.update.DefectUpdaterFactory インターフェイスを実装するクラス名に設定しなければなりません。

BTS Scanner の統合についての既存ドキュメントはこの章にあります。

---

## カスタム RMS Scanner

次の Java API を実装して、カスタムの RMS Scanner を作成することができます。

- com.parasoft.api.dtp.requirements.scan.RequirementScanner
- com.parasoft.api.dtp.requirements.scan.RequirementScannerFactory

Java API ドキュメントへのアクセスについては 864 ページの「Java API」を参照してください。ログに関する情報については 416 ページの「ログ ファイルの参照」を参照してください。

---

## カスタム実装のインストール

1. カスタム RMS Scanner/Updater のための jar ファイルを作成します。
2. この jar ファイルを \$DTP\_HOME/tomcat/webapps/grs/WEB-INF/lib ディレクトリにコピーします。
3. DTP を再起動します。

---

## カスタム実装を使用するよう RMS Scanner を構成する

\$DTP\_HOME/grs/config/bts に XML 構成ファイルを作成する必要があります。テンプレートとして \$DTP\_HOME/grs/config/bts/examples/ExampleCustomRmsScannerConfig.xml を使用できます。

scannerFactoryClass 要素の値は、  
com.parasoft.api.dtp.requirements.scan.RequirementScannerFactory インターフェイスを実装するクラス名に設定しなければなりません。これはカスタム XML 構成ファイルで最も重要な部分です。

## BTS Scanner の実行

「オリジナル ID の使用」の説明に従って BTS/RMS との統合を構成し、.xml ファイルを \$DTP\_HOME/grs/config/bts ディレクトリに保存すると、BTS Scanner がファイルを処理し、定義された BTS/RMS に接続して欠陥 / 要求に関する基本的な情報を DTP データベースにインポートします。

BTS Scanner は自動的に実行されますが、必要なときに手動で実行して BTS/RMS で行われた最新の変更を DTP のデータに反映することもできます。どちらの場合でも、BTS Scanner が期待どおりに動作していることを検証できます。

## BTS Scanner の自動実行

BTS Scanner は DTP のバックグラウンド ジョブです。BTS Scanner は 15 分ごとに実行されます。\$DTP\_HOME/grs/config/bts ディレクトリにある各構成ファイルで指定された BTS/RMS に接続し、データを DTP にインポートします。

BTS Scanner の実行間隔を設定することができます。次の操作を行います。

1. \$DTP\_HOME/grs/config/CronConfig.xml ファイルの frequency パラメーターを変更します。

```
<Job frequency="15" id="BTS Scanner" runDayOfWeek="*"
runHour="*" run-Minute="*">

    <class
name="com.parasoft.grs.rserver.cronjobs.bts.BtsScannerJob"
"/>

</Job>
```

2. DTP Server を再起動します。

## BTS Scanner の手動実行

必要なときに DTP の Web インターフェイスから BTS Scanner を実行できます。以下のどちらかの場所から実行できます。

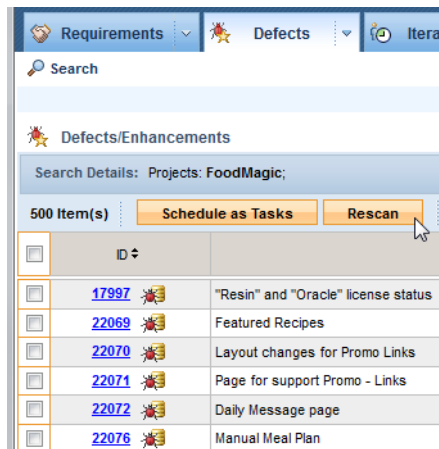
- Project Center: 欠陥 / 改善 の検索  
詳細については 345 ページの「[ 欠陥 / 改善 の検索 ] ページから BTS Scanner を手動で実行する」を参照してください。
- Report Center: メイン > 管理者 > ツール > 欠陥 / 改善の計算  
詳細については 590 ページの「スキャンと計算の自動実行」を参照してください。

### [ 欠陥 / 改善 の検索 ] ページから BTS Scanner を手動で実行する

1. Project Center で [ 欠陥 ] > [ 検索 ] をクリックし、検索を実行します。
2. [ 欠陥 / 改善 ] ページの [ 再スキャン ] ボタンをクリックします。

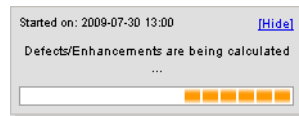


## 欠陥 / 改善の再スキャン



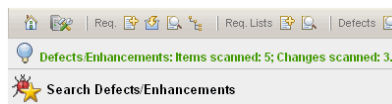
ダイアログが開き、スキャンの進捗が表示されます。

### スキャン進行中メッセージ



スキャンが完了すると、ステータス バーにスキャンされたバグおよび変更の数が表示されます。

### スキャン完了メッセージ



## BTS Scanner の動作確認

BTS Scanner は、自動的に実行することも、必要なときに手動で実行することもできます。ユーザーは、BTS Scanner が実行中かどうか、またデータが期待どおりに DTP にインポートされたかを検証できます。

BTS Scanner のバックグラウンド作業を検証するには、DTP\_HOME/logs ディレクトリにある `bts_scanner.log` ファイルを確認します (416 ページの「ログ ファイルの参照」を参照)。

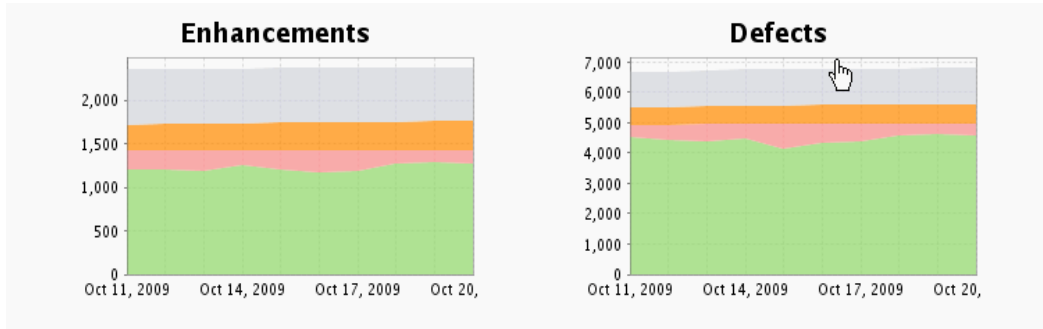
BTS Scanner が BTS/RMS からデータを DTP データベースにインポートしたかどうかを検証するには、以下のどちらかの操作を行います。

### 1. 欠陥 / 改善の検索

- [ 欠陥トラッキング システム ] ドロップダウン メニューを展開します。  
このメニューには DTP と統合されている BTS/RMS の名前が表示されます。名前は .xml 構成ファイルの `<bts/name>` ノードで定義した名前です。
- [ 検索 ] ボタンをクリックします。  
DTP データベースにインポートされたすべての欠陥が検索されて表示されます。

欠陥 / 改善の検索に関する詳細については『Project Center ユーザーズ ガイド』の「欠陥 / 改善の検索」を参照してください。

- Report Center で [ テスト ] > [ 変更ベースのテスト ] > [ 要求 / 欠陥 ] を選択します。
- デフォルト プロジェクトに切り替えます。DTP データベースのすべての欠陥 / 改善データが表示されます。



- Report Center と Project Center で欠陥 / 改善に対応する方法については、以下のセクションを参照してください。
  - 235 ページの「ソース管理フィルター」
  - 144 ページの「欠陥および改善レポート」
  - 93 ページの「欠陥 / 改善」
  - 106 ページの「要求と欠陥 / 改善のタスク化」

## BTS Updater の構成

DTP がサポートする BTS/RMS の場合、BTS/RMS 中の課題に関連付けられたタスクが変更されると、DTP は BTS/RMS 中のその課題のステータスとコメントを変更します。この処理は BTS Updater というユーティリティを使って行われます。BTS Updater は統合 / サポートされたバグトラッキング システムに DTP の更新を伝播します。

**注意：** 現在 BTS Updater はすべてのバージョンの Bugzilla でコメントの更新をサポートします。ただし、ステータスの更新ができるのは Bugzilla 3.4 以上だけです。

- サポートされる Bugzilla のバージョンについては 365 ページの「Bugzilla との統合」を参照してください。
- DTP がバグトラッキング システムとどのようにやり取りするのかについては 226 ページの「バグトラッキング システムとの同期」を参照してください。

BTS Updater を設定して、BTS/RMS にステータスとコメントの変更を伝播することができます。それには、BTS Scanner の .xml 構成ファイルを適切に設定します。この構成ファイルは、DTP Server がインストールされているホスト上の `DTP_HOME/grs/config/bts` ディレクトリにあります。

BTS Updater を有効化するには、新しいユーザー名とパスワードを既存の Bugzilla .xml 構成ファイルの `<connection-settings>` ノードに追加します。

例：

```

<bts>
  <connection-settings>
    <user>root</user>
    <pass encrypted="false">root</pass>
  </connection-settings>
</bts>

```

**重要！** このユーザーは、統合される BTS の Web インターフェイスに対するアカウントを持っていなければなりません。また、そのアカウントは BTS 中のすべてのプロジェクトに対して読み取り/書き込み権限を持つべきです。

---

## オリジナル ID の使用

[Requirements Code Review] および [Requirements Code Review Details] レポートで DTP の ID ではなくタスク / 要求のオリジナル ID を使用する場合、次の操作を行います。

1. `$DTP_HOME/grs/xreports/planning/req/RequirementsCodeReview.xml` ファイルを開きます。
2. `<private-parameters>` タグの下に次のエントリを追加します。  
`<parameter name="useOriginalId">true</parameter>`
3. `DTP_HOME/grs/xreports/codereview/CodeReviewDetails.xml` ファイルにも同じエントリを追加します。

---

## HP Quality Center

BTS Scanner は HP Quality Center (旧 Test Director) と連携するように構成できます。Test Director と Quality Center は同じ物を指しているため、このセクションではどちらも「QC」と呼びます。Quality Center の統合は、Quality Center 9、10、および 11 でテスト済みです。

BTS Scanner for QC を使用すると、QC データベースから Report Center に要求と欠陥をインポートできます。

DTP では 2 種類の方法で HP QC と統合することができます。

- HP QC データベースの直接アクセス。DTP は直接 QC データを読み取ってインポートします。
  - QC の要求は DTP に対する欠陥 (タイプは拡張) としてインポートされます。
  - QC の欠陥は DTP での欠陥としてインポートされます。
- オープン テスト アーキテクチャ (OTA) の使用。DTP は直接 QC データを読み取ってインポートします。
  - QC の要求は DTP に対する機能要求としてインポートされます。
  - QC の欠陥は DTP での欠陥としてインポートされます。

## HP QC データベースへの直接アクセス

**注意：**ほとんどのユーザーは、ステップ 2 を除く下記のすべてのステップで、デフォルト設定を利用できます。QC データベースの接続設定は、ユーザー独自の接続に合わせてカスタマイズする必要があります。

HP QC データベースへの直接アクセスを介して統合を構成するには、次の操作を行います。

1. 以下の全般的な設定を行います。

設定	説明
<bts>/<type>	"TestDirector" に設定するべきです。
<bts>/<name>	QC サーバーを識別するためにユーザーが見て分かる名前を指定します。 例： "Test Director" QC サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [ 欠陥 / 改善検索 ] ページに [ 欠陥トラッキング システム ] ドロップダウン リストの選択肢として表示されます。

例：

```
<bts type="TestDirector">
<name>Test Director</name>
```

2. QC データベース接続設定を指定します。

データは QC プロジェクトごとに別個のデータベースに保存されます。そのため、スキャン対象として少なくとも 1 つのデータベース名を指定する必要があります。

各プロジェクトのデータベース名は Quality Center Site Administration または MS SQL クライアントから確認できます。

**注意：**定義されている QC データベース名を記録しておくことを推奨します。データベース接続を設定するときに QC データベース名が必要です。

QC との統合では以下の QC データベース接続設定が必要です。

設定	説明
<bts>/<db-connection>/<db-type>	MSSQL のみ
<bts>/<db-connection>/<user>	データベース ユーザー名
<bts>/<db-connection>/<pass>	データベース パスワード
<bts>/<db-connection>/<url>	QC データベース接続 URL <b>注意：</b> connection url にデータベース名を含めないでください。
<bts>/<db-connection>/<database>	QC データベース名。少なくとも 1 つのデータベース名を指定します。

例：

```
<db-connection>
  <db-type>MSSQL</db-type>
  <!-- currently only MS SQL server supported. -->
  <user>sa</user>
  <!-- valid MS SQL server user -->
  <password encrypted="true">abc123</password>
  <!-- valid MS SQL server user passwd -->
  <url>jdbc:sqlserver://localhost;</url>
<!-- sample MSSQL connection url: <url>jdbc:sqlserver://HOST;</url> -->
```

**注意：** 接続 URL にはデータベース名を含めないでください。

```
<!-- At least one db name is required -->
<database>default_test_db</database>
<database>QualityCenter_Demo_db</database>
</db-connection>
```

### 3. フィールドマッピングを構成します。

欠陥と要求のフィールドマッピングは、データベースの BUG テーブルと REQ テーブルのカラム名に基づきます。各フィールドの詳細については、Quality Center のマニュアル (quality\_center\_db.chm) を参照してください。

欠陥および要求の field-mapping セクションはコメントアウトされています。セクションにはデフォルトのマッピング構成が記述されています。このセクションのコメントを削除して設定を変更できます。ただし、多くの場合、設定の変更は必要ありません。

**注意：** N/A はフィールドが利用できないことを表します。

例：

```
<defects>
  <fields-mapping>
    <summary>BG_SUMMARY</summary>
    <status>BG_STATUS</status>
    <resolution>BG_STATUS</resolution>
    <priority>BG_PRIORITY</priority>
    <severity>BG_SEVERITY</severity>
    <project>BG_PROJECT</project>
    <reporter>BG_DETECTED_BY</reporter>
    <assigned-to>BG_RESPONSIBLE</assigned-to>
    <creation-date>BG_DETECTION_DATE</creation-date>
    <version>BG_DETECTION_VERSION</version>
    <milestone>BG_PLANNED_CLOSING_VER</milestone>
    <hardware>N/A</hardware>
    <os>N/A</os>
    <component>BG_PROJECT</component>
    <modification-date>BG_VTS</modification-date>
  </fields-mapping>
</defects>

<requirements>
  <fields-mapping>
    <summary>RQ_REQ_NAME</summary>
    <status>RQ_REQ_STATUS</status>
```

```

        <resolution>RQ_REQ_STATUS</resolution>
        <priority>RQ_REQ_PRIORITY</priority>
        <severity>N/A</severity>
        <project>RQ_REQ_PRODUCT</project>
        <reporter>RQ_REQ_AUTHOR</reporter>
        <assigned-to>RQ_REQ_AUTHOR</assigned-to>
        <creation-date>RQ_REQ_DATE</creation-date>
        <version>N/A</version>
        <milestone>N/A</milestone>
        <hardware>N/A</hardware>
        <os>N/A</os>
        <component>RQ_REQ_TYPE</component>
        <modification-date>RQ_VTS</modification-date>
    </fields-mapping>
</requirements>

```

#### 4. 「解決時のステータス」と「アクティブでない解決」の値を指定します。

「解決時のステータス」と「アクティブでないステータス」のマッピング構成は、`fields-mapping.status` の表記法に依存する場合があります。ただし、通常は設定の変更は必要ありません。

例：

```

<resolved-status>
    <!-- Requirement -->
    <status>Passed</status>
    <status>Failed</status>
    <status>Reviewed</status>
    <status>Not Completed</status>

    <!-- Defects -->
    <status>Closed</status>
    <status>Fixed</status>
</resolved-status>

<inactive-resolution>
    <!-- Requirement -->
    <resolution>N/A</resolution>

    <!-- Defects -->
    <resolution>Rejected</resolution>
</inactive-resolution>

```

#### 注意：

- Report Center にインポートされた後、スキャンされた QC 項目に次の書式で識別子が付けられます。
  - PR の場合：BUG\_( 実際の QC ID)
  - FR の場合：REQ\_( 実際の QC ID)
- 現時点では、Report Center のレポートから QC の PR/FR 詳細を開くリンク機能は使用できません。
- Task Activity レポートにデータを表示するには、QC ユーザーを Report Center データベースに追加する必要があります。

- これはインクリメンタル スキャンです。つまり、前回のスキャンの後に QC で変更された要求 / 欠陥だけが DTP に対して更新されます。どの項目が QC で変更されたかを DTP が確実に把握できるようにするには、QC の Project Customization/Project Entities フォームで適切な設定を行う必要があります。ステップ 3 の「フィールド マッピングの構成」に記載されたすべてのフィールドは、アクティブな History プロパティを QC で持つべきです。

## Windows 認証を使った Quality Center データベースへの接続

Windows 認証を介して MSSQL サーバーに接続するには、Windows 環境に DTP サービスがインストールされていて実行中でなければなりません。なぜなら、公式の MSSQL Server JDBC ドライバが使用する証明書は、Windows 認証のためのサービスを現在実行しているユーザー アカウントからの証明書だけだからです。

DTP のセットアップのために準備するには、次の操作を行います。

1. 公式の MSSQL Server JDBC ドライバ パッケージをダウンロードし、操作指示に従ってパッケージをアンロードします。
2. Windows のコントロール パネルを開き、[ システムとセキュリティ ] > [ 管理ツール ] をクリックします。
3. [ サービス ] アイコンをダブルクリックします。
4. Parasoft Development Testing Platform を右クリックしてショートカット メニューの [ プロパティ ] をクリックします。
5. [ ログオン ] タブをクリックします。[ アカウント ] を選択し、LOCATION\username の書式で Windows ユーザー アカウントを入力します。このユーザー アカウントは MSSQL サーバー データベースにアクセスできなければなりません。
6. パスワードを入力し、[ OK ] をクリックします。
7. Parasoft Development Testing Platform を選択した状態で [ サービスの停止 ] リンクをクリックし、DTP のサービスを停止します。
8. 新しいウィンドウで `${DTP_HOME}/tomcat/lib/thirdparty` ディレクトリを開きます。
9. `sqljdbc4.jar` ファイルのバックアップ コピーを作成します。
10. `<MSSQL_JDBC_DRIVER>\sqljdbc_<VERSION>\<LANG>` の `sqljdbc4.jar` を現行ディレクトリにコピーします。
11. `${DTP_HOME}/jre/bin` ディレクトリにアクセスし、`<MSSQL_JDBC_DRIVER>\sqljdbc_<VERSION>\<LANG>\auth\<ARCHITECTURE>` ディレクトリの `sqljdbc_auth.dll` を追加します。

DTP をセットアップするには、次の操作を行います。

1. `${DTP_HOME}/grs/config/bts` ディレクトリにアクセスし、QC BTS Scanner 構成ファイルを開きます。
2. QC BTS Scanner 構成ファイル中で `<db-connection>` 要素を探します。この構成ファイルがまだ設定されていない場合、357 ページの「HP QC データベースへの直接アクセス」の説明に従って適切に構成ファイルを変更します。

3. <db-connection> 要素内の <url> 要素の URL 値に対して、プロパティ `integratedSecurity=true` を追加します。
4. 以下の例のように設定して変更を保存します。

```
<db-connection>

<db-type>MSSQL</db-type>
<user>sa</user>
<password encrypted="true">abc123</password>
<url>jdbc:sqlserver://localhost;integratedSecurity=true;</url>

</db-connection>
```

5. [ サービスの開始 ] リンクをクリックしてサービスの一覧から DTP サービスを開始します。
6. 346 ページの「BTS Scanner の動作確認」の説明に従って、QC Scanner が適切に構成されたことを確認します。

## OTA (Open Test Architecture) の使用

OTA API を使用する BTS Scanner for QC を利用すると、QC データベースから DTP に要求と欠陥をインポートすることができます。QC の要求は Project Center の要求として、そして QC の欠陥は Project Center のバグとして表示されます。インポートされた要求と欠陥は読み取り専用モードであり、そのほとんどの属性を変更できません。

DTP がインストールされているのと同じマシンに QC クライアントがインストールされている場合、DTP から取得したデータを使って QC の要求を更新することもできます。

OTA (Open Test Architecture) を使って統合を構成するには、次の操作を行います。

1. 以下の全般的な設定を行います。

設定	説明
<bts>/<type>	"TestDirectorOTA" に設定するべきです。
<bts>/<name>	QC サーバーを識別するためにユーザーが見て分かる名前を指定します。 例: "Test Director"  QC サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [ 欠陥 / 改善検索 ] ページに [ 欠陥トラッキングシステム ] ドロップダウン リストの選択肢として表示されます。

例:

```
<bts type="TestDirectorOTA">
<name>QC OTA</name>
```

Quality Center データベース接続設定を指定します。

データは QC プロジェクトごとに別個のデータベースに保存されます。このため、スキャン



対象として少なくとも 1 つのデータベース名を指定する必要があります (下記のサンプルの <database> タグを参照)。

各プロジェクトのデータベース名は Quality Center Site Administration または MS SQL クライアントから確認できます。

**注意：** 定義されている QC データベース名を書きとめておくことを推奨します。データベース接続を設定するときにこの情報が必要になります。

QC との統合では以下の QC データベース接続設定が必要です。

設定	説明
<bts>/<db-connection>/<db-type>	データベース ベンダー：MS SQL のみ。
<bts>/<db-connection>/<user>	データベース ユーザー。
<bts>/<db-connection>/<pass>	データベース パスワード。
<bts>/<db-connection>/<url>	Quality Center データベース接続 URL。 <b>注意：</b> データベース接続 URL にはデータベース名を含めないでください。
<bts>/<db-connection>/<database>	QC データベース名。少なくとも 1 つのデータベース名が必要です。 さらに <database> タグの次の属性を指定する必要があります。 <ul style="list-style-type: none"> <li>• qc-domain – qc-project の QC ドメインを指定します。</li> <li>• qc-project – QC プロジェクトの名前を指定します。</li> <li>• dtp-project – DTP プロジェクトの名前を指定します。スキャンされた要求はこのプロジェクトの下に格納されます。</li> </ul> <b>注意：</b> スキャン プロセス中、指定されたプロジェクトが DTP データベース中にない場合、プロジェクトは自動的に作成されます。要求を適切に作成するために、qc-domain、qc-project、および dtp-project 属性が使用されます。

例：

```
<db-connection>
<db-type>MSSQL</db-type>
<user>sa</user>
<password encrypted="true">abc123</password>
<url>jdbc:sqlserver://localhost;</url>
<database qc-domain="DEFAULT" qc-project="QualityCenter_Demo"
dtp-project="MyProject">QualityCenter_Demo_db
</database>
</db-connection>
```

## 2. ota-connection を設定します。

DTP のデータで QC の欠陥を自動的に更新できるようにするには、OTA API を使ったやり取りを可能にする証明書を指定する必要があります。それには、以下の全般的な設定を行います。

設定	説明
<bts>/<ota-connection>/<user>	OTA API を使ってやり取りできるユーザーの名前。
<bts>/<ota-connection>/<password>	指定されたユーザーのための有効なパスワード。
<bts>/<ota-connection>/<url>	QC への有効な HTTP URL。

例：

```
<ota-connection>
<user>sa</user>
<password>sa</password>
<url>http://localhost:8080/qcbin</url>
</ota-connection>
```

# Report Center および Project Center と サードパーティ製ツールの統合

このセクションの内容：

- バグ トラッキング システムおよび要求管理システムの統合
- コード レビューの統合
- Report Center と Emma の統合
- カスタム プロセッサとの統合
- Report Center とソース管理システムの統合
- サードパーティのツールから DTP へのテスト結果の送信

# HP Quality Center との統合

BTS Scanner を設定して HP Quality Center (以前の Test Director) と連携させることができます。Test Director と Quality Center は同義であるため、このセクションでは両方を指して "QC" と呼びます。Quality Center の統合は、Quality Center 9、10、および 11 でテスト済みです。

BTS Scanner for QC を使用すると、QC データベースから Report Center に要求と欠陥をインポートすることができます。

Development Testing Platform (DTP) では 2 種類の方法で HP QC と統合することができます。

- HP QC データベースの直接アクセス。DTP は直接 QC データを読み取ってインポートします。
  - QC の要求は DTP に対する機能要求としてインポートされます。
  - QC の欠陥は DTP での欠陥としてインポートされます。
- オープン テスト アーキテクチャ (OTA) の使用。DTP は直接 QC データを読み取ってインポートします。
  - QC の要求は DTP に対する機能要求としてインポートされます。
  - QC の欠陥は DTP での欠陥としてインポートされます。

## HP QC データベースへの直接アクセス

**注意:** ほとんどのユーザーは、ステップ 2 を除く下記のすべてのステップで、デフォルト設定を利用できます。QC データベースの接続設定は、ユーザー独自の接続に合わせてカスタマイズする必要があります。

HP QC データベースへの直接アクセスによって統合を設定するには、次の操作を行います。

1. 以下の全般的な設定を行います。

設定	説明
<bts>/<type>	"TestDirector" に設定するべきです。
<bts>/<name>	HP QC サーバーを識別するためにユーザーが見て分かる名前を指定します。 例: "Test Director" HP QC サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [欠陥 / 改善検索] ページに [欠陥トラッキング システム] ドロップダウン リストの選択肢として表示されます。

例:

```
<bts type="TestDirector">
<name>Test Director</name>
```

2. QC データベース接続設定を指定します。

データは QC プロジェクトごとに別個のデータベースに保存されます。そのため、スキャン対象として少なくとも 1 つのデータベース名を指定する必要があります。

各プロジェクトのデータベース名は Quality Center Site Administration または MS SQL クライアントから確認できます。

**注意：**定義されている QC データベース名を記録しておくことを推奨します。データベース接続を設定するときに QC データベース名が必要になります。

QC との統合では以下の QC データベース接続設定が必要です。

設定	説明
<bts>/<db-connection>/<db-type>	MSSQL のみ
<bts>/<db-connection>/<user>	データベース ユーザー名
<bts>/<db-connection>/<pass>	データベース パスワード
<bts>/<db-connection>/<url>	QC データベース接続 URL <b>注意：</b> connection url にデータベース名を含めないでください。
<bts>/<db-connection>/<database>	QC データベース名。少なくとも 1 つのデータベース名を指定します。

例：

```
<db-connection>
  <db-type>MSSQL</db-type>
  <!-- currently only MS SQL server supported. -->
  <user>sa</user>
  <!-- valid MS SQL server user -->
  <password encrypted="true">abc123</password>
  <!-- valid MS SQL server user passwd -->
  <url>jdbc:sqlserver://localhost;</url>
<!-- sample MSSQL connection url: <url>jdbc:sqlserver://HOST;</url> -->
```

**注意：**接続 URL にデータベース名を含めてはいけません。

```
<!-- At least one db name is required -->
<database>default_test_db</database>
<database>QualityCenter_Demo_db</database>
</db-connection>
```

### 3. フィールドマッピングを構成します。

欠陥と要求のフィールドマッピングは、データベースの BUG テーブルと REQ テーブルのカラム名に基づきます。各フィールドの詳細については、Quality Center のマニュアル (quality\_center\_db.chm) を参照してください。

欠陥および要求の field-mapping セクションはコメントアウトされています。セクションにはデフォルトのマッピング構成が記述されています。このセクションのコメントを削除して設定を変更できます。ただし、多くの場合、設定の変更は必要ありません。

**注意：** N/A はフィールドが利用できないことを表します。

例：

```
<defects>
  <fields-mapping>
    <summary>BG_SUMMARY</summary>
    <status>BG_STATUS</status>
    <resolution>BG_STATUS</resolution>
    <priority>BG_PRIORITY</priority>
    <severity>BG_SEVERITY</severity>
    <project>BG_PROJECT</project>
    <reporter>BG_DETECTED_BY</reporter>
    <assigned-to>BG_RESPONSIBLE</assigned-to>
    <creation-date>BG_DETECTION_DATE</creation-date>
    <version>BG_DETECTION_VERSION</version>
    <milestone>BG_PLANNED_CLOSING_VER</milestone>
    <hardware>N/A</hardware>
    <os>N/A</os>
    <component>BG_PROJECT</component>
    <modification-date>BG_VTS</modification-date>
  </fields-mapping>
</defects>

<requirements>
  <fields-mapping>
    <summary>RQ_REQ_NAME</summary>
    <status>RQ_REQ_STATUS</status>
    <resolution>RQ_REQ_STATUS</resolution>
    <priority>RQ_REQ_PRIORITY</priority>
    <severity>N/A</severity>
    <project>RQ_REQ_PRODUCT</project>
    <reporter>RQ_REQ_AUTHOR</reporter>
    <assigned-to>RQ_REQ_AUTHOR</assigned-to>
    <creation-date>RQ_REQ_DATE</creation-date>
    <version>N/A</version>
    <milestone>N/A</milestone>
    <hardware>N/A</hardware>
    <os>N/A</os>
    <component>RQ_REQ_TYPE</component>
    <modification-date>RQ_VTS</modification-date>
  </fields-mapping>
</requirements>
```

#### 4. 「解決時のステータス」と「アクティブでない解決」の値を指定します。

「解決時のステータス」と「アクティブでないステータス」のマッピング構成は、fields-mapping.status の表記法に依存する場合があります。ただし、通常は設定の変更は必要ありません。

例：

```
<resolved-status>
  <!-- Requirement -->
  <status>Passed</status>
  <status>Failed</status>
```

```

<status>Reviewed</status>
<status>Not Completed</status>

<!-- Defects -->
<status>Closed</status>
<status>Fixed</status>
</resolved-status>

<inactive-resolution>
<!-- Requirement -->
<resolution>N/A</resolution>

<!-- Defects -->
<resolution>Rejected</resolution>
</inactive-resolution>

```

**注意：**

- Report Center にインポートされた後、スキャンされた QC 項目に次の書式で識別子が付けられます。
  - PR の場合：BUG\_( 実際の QC ID)
  - FR の場合：REQ\_( 実際の QC ID)
- 現時点では、Report Center のレポートから QC の PR/FR 詳細を開くリンク機能は使用できません。
- Task Activity レポートにデータを表示するには、QC ユーザーを Report Center データベースに追加する必要があります。
- これはインクリメンタル スキャンです。つまり、前回のスキャンの後に QC で変更された要求 / 欠陥だけが DTP に対して更新されます。どの項目が QC で変更されたかを DTP が確実に把握できるようにするには、QC の Project Customization/Project Entities フォームで適切な設定を行う必要があります。ステップ 3 の「フィールド マッピングの構成」に記載されたすべてのフィールドは、アクティブな History プロパティを QC で持つべきです。

**Windows 認証を使った Quality Center データベースへの接続**

Windows 認証を介して MSSQL サーバーに接続するには、Windows 環境に DTP サービスがインストールされていて実行中でなければなりません。なぜなら、公式の MSSQL Server JDBC ドライバが使用する証明書は、Windows 認証のためのサービスを現在実行しているユーザー アカウントからの証明書だけだからです。

DTP のセットアップのために準備するには、次の操作を行います。

1. 公式の MSSQL Server JDBC ドライバ パッケージをダウンロードし、操作指示に従ってパッケージをアンロードします。
2. [ スタート ] メニューを開き、[ コンピューター ] を右クリックして [ 管理 ] をクリックします ( 別の方法としては、コントロール パネルを開き、[ システムとセキュリティ ] > [ 管理ツール ] をクリックします )。
3. [ サービスとアプリケーション ] を展開し、[ サービス ] を選択して Windows サービスの一覧を表示します。
4. Parasoft Development Testing Platform を右クリックして [ プロパティ ] を選択します。

5. [ ログオン ] タブをクリックします。[ アカウント ] を選択し、LOCATION\username の書式で Windows ユーザー アカウントを入力します。このユーザー アカウントは MSSQL サーバー データベースにアクセスできなければなりません。
6. パスワードを入力し、[OK] をクリックします。
7. Parasoft Development Testing Platform サービスを選択した状態で [ サービスの停止 ] リンクをクリックし、DTP のサービスを停止します。
8. 新しいウィンドウで \${DTP\_HOME}/tomcat/lib ディレクトリを開きます。
9. sqljdbc4.jar ファイルのバックアップ コピーを作成します。
10. <MSSQL\_JDBC\_DRIVER>\sqljdbc\_<VERSION>\<LANG> の sqljdbc4.jar を現行ディレクトリにコピーします。
11. \${DTP\_HOME}/jre/bin ディレクトリにアクセスし、<MSSQL\_JDBC\_DRIVER>\sqljdbc\_<VERSION>\<LANG>\auth\<ARCHITECTURE> ディレクトリの sqljdbc\_auth.dll を追加します。

DTP をセットアップするには、次の操作を行います。

1. \${DTP\_HOME}/grs/config/bts ディレクトリにアクセスし、QC BTS Scanner 構成ファイルを開きます。
2. QC BTS Scanner 構成ファイル中で <db-connection> 要素を探します。この構成ファイルがまだ設定されていない場合、357 ページの「HP QC データベースへの直接アクセス」の説明に従って適切に構成ファイルを変更します。
3. <db-connection> 要素内の <url> 要素の URL 値に対して、プロパティ integratedSecurity=true を追加します。
4. 以下の例のように設定して変更を保存します。

```
<db-connection>

<db-type>MSSQL</db-type>
<user>sa</user>
<password encrypted="true">abc123</password>
<url>jdbc:sqlserver://localhost;integratedSecurity=true;</url>

</db-connection>
```

5. [ サービスの開始 ] リンクをクリックしてサービスの一覧から Parasoft Development Testing Platform サービスを開始します。
6. 346 ページの「BTS Scanner の動作確認」の説明に従って、QC Scanner が適切に構成されたことを確認します。



## OTA (Open Test Architecture) の使用

OTA API を使用する BTS Scanner for QC を利用すると、QC データベースから DTP に要求と欠陥をインポートすることができます。QC の要求は Project Center の要求として、そして QC の欠陥は Project Center のバグとして表示されます。インポートされた要求と欠陥は読み取り専用モードであり、そのほとんどの属性を変更できません。

DTP がインストールされているのと同じマシンに QC クライアントがインストールされている場合、DTP から取得したデータを使って QC の要求を更新することもできます。

OTA (Open Test Architecture) を使って統合を構成するには、次の操作を行います。

1. 以下の一般的な設定を行います。

設定	説明
<bts>/<type>	"TestDirectorOTA" に設定するべきです。
<bts>/<name>	QC サーバーを識別するためにユーザーが見て分かる名前を指定します。 例: "Test Director"  QC サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [ 欠陥 / 改善検索 ] ページに [ 欠陥トラッキングシステム ] ドロップダウン リストの選択肢として表示されます。

例:

```
<bts type="TestDirectorOTA">
<name>QC OTA</name>
```

2. Quality Center データベース接続設定を指定します。

データは QC プロジェクトごとに別個のデータベースに保存されます。このため、スキャン対象として少なくとも 1 つのデータベース名を指定する必要があります。

各プロジェクトのデータベース名は Quality Center Site Administration または MS SQL クライアントから確認できます。

**注意:** 定義されている QC データベース名を書きとめておくことを推奨します。データベース接続を設定するときにこの情報が必要になります。

QC との統合では以下の QC データベース接続設定が必要です。

設定	説明
<bts>/<db-connection>/<db-type>	データベース ベンダー: MS SQL のみ。
<bts>/<db-connection>/<user>	データベース ユーザー。
<bts>/<db-connection>/<pass>	データベース パスワード。

設定	説明
<bts>/<db-connection>/<url>	Quality Center データベース接続 URL。 <b>注意：</b> データベース接続 URL にはデータベース名を含めないでください。
<bts>/<db-connection>/<database>	QC データベース名。少なくとも 1 つのデータベース名が必要です。 さらに <database> タグの次の属性を指定する必要があります。 <ul style="list-style-type: none"> <li>• qc-domain – qc-project の QC ドメインを指定します。</li> <li>• qc-project – QC プロジェクトの名前を指定します。</li> <li>• dtp-project – DTP プロジェクトの名前を指定します。スキャンされた要求はこのプロジェクトの下に格納されます。</li> </ul> <b>注意：</b> スキャン プロセス中、指定されたプロジェクトが DTP データベース中にない場合、プロジェクトは自動的に作成されます。要求を適切に作成するために、qc-domain、qc-project、および dtp-project 属性が使用されます。

例：

```
<db-connection>
<db-type>MSSQL</db-type>
<user>sa</user>
<password encrypted="true">abc123</password>
<url>jdbc:sqlserver://localhost;</url>
<database qc-domain="DEFAULT" qc-project="QualityCenter_Demo"
dtp-project="MyProject">QualityCenter_Demo_db
</database>
</db-connection>
```

### 3. ota-connection を設定します。

Development Testing Platform のデータで QC の欠陥を自動的に更新できるようにするには、OTA API を使ったやり取りを可能にする証明書を指定する必要があります。それには、以下の全般的な設定を行います。

設定	説明
<bts>/<ota-connection>/<user>	OTA API を使ってやり取りできるユーザーの名前。
<bts>/<ota-connection>/<password>	指定されたユーザーのための有効なパスワード。

設定	説明
<bts>/<ota-connection>/ <url>	QC への有効な HTTP URL。

例：

```
<ota-connection>  
<user>sa</user>  
<password>sa</password>  
<url>http://localhost:8080/qcbin</url>  
</ota-connection>
```

# Bugzilla との統合

Bugzilla との統合は、以下のバージョンでテスト済みです。

- 2.16
- 2.18
- 2.20
- 2.22
- 3.0
- 3.4
- 3.6
- 4.0
- 4.2

BTS Updater は Bugzilla 3.4 以降で動作します。

ビルトイン UI を使って Bugzilla と統合する方法については 347 ページの「BTS Updater の構成」を参照してください。

Bugzilla のスキャンを設定するには、DTP\grs\config\bts\bugzila.xml ファイルで以下の設定をカスタマイズする必要があります。

1. 以下の全般的な設定を指定します。

設定	説明
<bts>/<name>	Bugzilla サーバーを識別するためにユーザーが見て分かる名前を指定します (たとえば My Bugzilla Server)。Bugzilla サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [ 欠陥 / 改善検索 ] ページで [ 欠陥トラッキング システム ] ドロップダウン リストの選択肢として表示されます。
<bts>/<url-prefix>	Report Center サーバーから Bugzilla サーバーへのバグ詳細リンクを作成するために必要な接頭辞を指定します。
<bts>/<version>	DTP に統合する Bugzilla のバージョンを指定します。

2. 以下の Bugzilla データベース接続設定を指定します。

設定	説明
<bts>/<db-connection>/<db-type>	データベース ベンダー。MySQL または Oracle を指定します。
<bts>/<db-connection>/<user>	データベース ユーザー名

設定	説明
<bts>/<db-connection>/<pass>	データベース パスワード
<bts>/<db-connection>/<url>	Bugzilla データベース接続 URL

例：

```
<bts type="Bugzilla">
  <name>My Bugzilla Server</name>
  <url-prefix>http://mybugzillaserver/bugzilla/
show_bug.cgi?id=</url-prefix>
  <version>3.4</version>
  <db-connection>
    <db-type>MySQL</db-type>
    <user>bugs</user>
    <password encrypted="true">abc123</password>
    <url>jdbc:mysql://bugzilla.host.com:3306/bugs</url>
<!-- sample Oracle connection url: <url>jdbc:oracle:thin:@HOST:PORT:SID/<
url> -->
  <!-- sample MySQL connection url: <url>jdbc:mysql://HOST:PORT/DATABASE/<
url> -->
  </db-connection>
</bts>
```

### 3. 「解決済みステータス」および「非アクティブの解決」の値を指定します。

言い換えると、バグが修正されたことを表すステータスと、バグがアクティブでないことを表すステータスを指定します。これらの情報がないと、Report Center はバグ履歴レポートを作成できません。

例：

```
<resolved-status>
  <status>VERIFIED</status>
  <status>RESOLVED</status>
  <status>CLOSED</status>
</resolved-status>

<inactive-resolution>
  <resolution>INVALID</resolution>
  <resolution>WONTFIX</resolution>
  <resolution>DUPLICATE</resolution>
  <resolution>MOVED</resolution>
  <resolution>REMIND</resolution>
  <resolution>LATER</resolution>
  <resolution>WORKSFORME</resolution>
</inactive-resolution>
```

### 4. <feature-request> 設定を使って、バグが改善であることを表す、Bugzilla の重要度フィールドの値を指定します。改善を表す重要度の値は Bugzilla のバージョンによって異なります。

Bugzilla バージョン 3.0 以前の場合、改善を表す重要度の値は "Feature\_Req" です。構成は次のようになります。

```
<feature-request>
  <item>Feature_Req</item>
</feature-request>
```

Bugzilla バージョン 3.4 の場合、改善を表す重要度の値は "enhancement" です。構成は次のようになります。

```
<feature-request>
  <item>enhancement</item>
</feature-request>
```

# IBM Rational ClearQuest との統合

ClearQuest の統合は、以下のバージョンでテスト済みです。

- 2003.06.15.734.000
- 2007
- 7.0.1.1

ClearQuest と DTP を統合する前に、DTP がインストールされているのと同じマシンに ClearQuest クライアント プログラム (cqperl コマンド) がインストールされていることを確認してください。BTS Scanner は cqperl コマンドを使って ClearQuest から項目を読み込みます。

以下のディレクトリにある ExampleClearQuestScannerConfig.xml ファイルのコピーを作成します。

```
{DTP_HOME}\grs\config\bts\examples\
```

ClearQuest のスキャンを設定するには、コピーしたファイルに対して以下の設定をカスタマイズする必要があります。

1. 以下の全般的な設定を指定します。

設定	説明
<bts>/<name>	ClearQuest サーバーを識別するためにユーザーが見て分かる名前を指定します (たとえば My ClearQuest Defects Server)。ClearQuest サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [ 欠陥 / 改善検索 ] ページで [ 欠陥トラッキング システム ] ドロップダウン リストの選択肢として表示されます。
<bts>/<connection-settings>/<dbset>	ClearQuest データベース セット。
<bts>/<connection-settings>/<dbname>	ClearQuest データベース名。
<bts>/<connection-settings>/<user>	ユーザー名。
<bts>/<connection-settings>/<pass>	ユーザー パスワード。

2. <run-options> を使って、BTS Scanner で読み込む ClearQuest エンティティ名を定義します。

設定	説明
<run-options>/<entity-type>	インポート対象の項目（欠陥、変更リクエストなど）に関する情報を保持する ClearQuest データベースのエンティティ名。BTS Scanner はこのエンティティで指定された項目だけをスキャンします。

**注意：** ClearQuest からインポートしたい <entity-type> ごとに個別に BTS Scanner 構成ファイルを用意する必要があります。

3. (オプション) ClearQuest のエンティティの一部またはすべてを、DTP で改善として扱うかどうかを指定します。

ClearQuest のエンティティの一部またはすべてを、欠陥または改善のどちらとしてインポートするかを指定できます。デフォルトでは、ClearQuest の項目は欠陥として扱われます。

BTS Scanner は、インポートされたエンティティの severity フィールドの値に基づいてエンティティを改善として扱うかどうかを決定します。

ClearQuest のエンティティの一部またはすべてを改善として扱う場合、次の操作を行います。

- 「サンプル構成ファイルの準備」および「構成ファイルの追加」の説明に従って BTS Scanner 構成ファイルを準備します。
- 改善としてインポートするエンティティタイプのために <entity-type> を定義し、Clear Quest でのエンティティ名を指定します。

たとえば、ClearQuest では変更リクエストが "cr" エンティティとして保存されている場合、次のように <entity-type> を定義します。

```
<run-options>
  <entity-type>cr</entity-type>
</run-options>
```

- 以下のどちらかの操作を行います。

ClearQuest のエンティティの一部を改善として扱う場合：

- <feature-request> タグを使って、ClearQuest から読み取られたどの項目を改善として扱うかを指定します。

例：

```
<feature-request>/<item> 6-Enhancement </feature-request>
```

上記のように構成すると、Severity フィールドの値が "6-Enhancement" の項目はすべて改善として DTP にインポートされ、他の項目は欠陥としてインポートされます。

すべての ClearQuest エンティティを改善として扱うには、次のように構成します。

- <feature-request> タグに、取り得るすべての Severity フィールド値を指定します。



例：

```
<feature-request>
  <item>1-Critical</item>
  <item>2-High Attention</item>
  <item>3 -Major</item>
  <item>4-Average</item>
  <item>5-Minor</item>
  <item>6-Enhancement</item>
</feature-request>
```

#### 4. ClearQuest データベース フィールドマッピングを指定します。

BTS Scanner は、Report Center のフィールドにマッピングされた ClearQuest のバグフィールドだけをインポートします。マッピングされていないフィールドは不明なフィールドとして扱われ、変換プロセス中にスキップされます。

**注意：**どのテーブル フィールドが自分の ClearQuest データベース中にあるか分からない場合、read\_single\_bug.pl スクリプトを実行して、ClearQuest データベースの 1 つのサンプル項目のテーブル フィールドを出力できます。このスクリプトは DTP\grs\extras\ClearQuest にあります。

このスクリプトを実行する前に、スクリプト ファイルを開いてログイン情報とサンプルの欠陥 ID を編集してください。

たとえば、ClearQuest の Headline フィールドは Report Center の Summary フィールドに対応し、Submitter フィールドは Reporter フィールドに対応するといったマッピングを指定します。

一部のフィールドは ClearQuest との統合に必ず必要ですが、他のフィールドはオプションです。

以下のフィールドマッピング設定が必要です。

フィールド	説明
<bts>/<fields-mapping>/<summary>	バグの概要。
<bts>/<fields-mapping>/<status>	現在のステータス。open、closed など。
<bts>/<fields-mapping>/<resolution>	バグの解決方法。fixed、wontfix など。
<bts>/<fields-mapping>/<priority>	バグの優先度。low、medium など。
<bts>/<fields-mapping>/<severity>	バグの重要度。minor、critical など。
<bts>/<fields-mapping>/<reporter>	バグの報告者。
<bts>/<fields-mapping>/<creation-date>	バグの作成日。

以下のフィールドマッピング設定はオプションです。

フィールド	説明
<bts>/<fields-mapping>/ <project>	バグが属するプロジェクト。
<bts>/<fields-mapping>/ <version>	プロジェクトのバージョン。
<bts>/<fields-mapping>/ <milestone>	プロジェクトのマイルストーン名。
<bts>/<fields-mapping>/ <hardware>	バグの影響を受けるハードウェア。
<bts>/<fields-mapping>/ <os>	バグの影響を受ける OS。
<bts>/<fields-mapping>/ <component>	プロジェクト コンポーネント。
<bts>/<fields-mapping>/ <modification-date>	バグの最終更新日。
<bts>/<fields-mapping>/ <assigned-to>	現在バグが割り当てられているユーザー。

例：

```
<fields-mapping>
  <summary>Headline</summary> *
  <status>State</status> *
  <resolution>Resolution</resolution> *
  <priority>Priority</priority> *
  <severity>Severity</severity> *
  <reporter>Submitter</reporter> *
  <creation-date>Submit_Date</creation-date> *
  <project>Project</project>
  <version>Version</version>
  <milestone>Milestone</milestone>
  <hardware>Hardware</hardware>
  <os>OS</os>
  <component>Component</component>
  <modification-date>Modification_Date</modification-date>
  <assigned-to>Owner</assigned-to>
</fields-mapping>
```

**注意：**アスタリスクが付いている項目は必須です。必須ではないフィールドを指定した場合、それらのフィールドの値に基づいて Report Center のプロジェクトにフィルターを追加することができます。

5. 「解決済みステータス」 および 「非アクティブの解決」 の値を指定します。

言い換えると、バグが修正されたことを表すステータスと、バグがアクティブでないことを

表すステータスを指定します。これらの情報がないと、Report Center はバグ履歴レポートを作成できません。

例：

```
<resolved-status>
  <status>Resolved</status>
  <status>Closed</status>
</resolved-status>

<inactive-resolution>
  <resolution>Duplicate</resolution>
  <resolution>Functions as Designed</resolution>
</inactive-resolution>
```

## 6. 以下の構成も必要です。

- <date-formats>

ClearQuest で使われている日付書式が以下のいずれかであることを確認してください。以下のリストにない場合、書式を追加してください。この情報は、DTP が ClearQuest から項目を読み込むときに、正しく日付を解釈するために必要です。

デフォルトの構成は次のとおりです。

```
<date-formats>
  <date-format>MMM-dd-yyyy HH:mm</date-format>
  <date-format>MM/dd/yyyy h:mm a</date-format>
  <date-format>yyyy-MM-dd HH:mm:ss</date-format>
</date-formats>
```

**注意：**書式は Java の SimpleDateFormat クラスに定義された書式のいずれかでなければなりません ( <http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html> を参照 )。

- <bugs-to-read>

この設定は、テスト目的でのみ使用されます。設定をテストするときに、何個の項目をインポートしたら BTS Scanner が停止するかを指定できます。たとえば 10 を指定すると、BTS Scanner は 10 個の項目をインポートした後に停止します。

**注意：**0 を指定すると、すべての項目が ClearQuest から読み込まれます。

- <retries-on-error>

まれに BTS Scanner からの cqperl の呼び出しが ClearQuest への接続に失敗する場合があります。この設定を指定すると、BTS Scanner は接続の問題があったときに cqperl の呼び出しを再試行します。

- <scan-changed-from>, <scan-changed-to>

この設定を指定すると、指定した期間に変更された項目だけがスキャンされます。

---

# Atlassian JIRA との統合

JIRA との統合は以下のバージョンでテスト済みです。

- 3.6.2
- 3.7.3
- 3.10.1
- 3.10.2
- 3.12
- 3.13
- 4.1.1
- 4.1.2
- 4.3
- 4.4
- 5.x
- 6.x.

BTS Scanner と JIRA の連携を構成するには、JIRA 側と DTP 側の双方でいくつかの設定を行う必要があります。必要な調整について以下のセクションで説明します。

- JIRA サーバー側での設定
- DTP 側での JIRA の設定

BTS Scanner は HTTPS/SSL 経由で JIRA と連携するよう構成することもできます。詳細な方法については次のセクションを参照してください。

- HTTPS/SSL 経由で BTS Scanner と JIRA を連携するための構成

---

## 5.x および 6.x との統合

JIRA 5.x および 6.x との統合にはビルトイン UI を使用します (336 ページの「ビルトイン UI を使った BTS/RMS Scanner の構成」を参照)。JIRA 5.x および 6.x と統合するには、DTP Server で SSL を有効化する必要があります。詳細については 758 ページの「SSL を使用した Development Testing Platform の実行」を参照してください。

---

## JIRA 4.4 以前との統合

以下のセクションでは、JIRA 4.4 以前の設定について説明します。ビルトイン UI を使って JIRA 4.x に対して DTP を設定できますが、JIRA サーバー側でも設定が必要です。このセクションの説明を参照してください。

すでに JIRA 4.x scanner を設定済みである場合 (たとえば 4.9.3 より前の DTP で JIRA 4.x scanner を設定済みである場合)、DTP の UI からさらに JIRA 5.x scanner を設定することはできません。これは制限事項ではなく、DTP で JIRA の課題が重複するのを防ぐための制約です。

既存の JIRA 4.x scanner を JIRA 5.x scanner に移行したい場合は Parasoft Support に連絡してください。移行を可能にするためにデータベースを手動で更新する必要があります。

## JIRA サーバー側での設定

JIRA を設定する前に、JIRA サーバーで SOA インターフェイスが有効化されていることを確認する必要があります。JIRA リモート インターフェイスの有効化については、JIRA のドキュメントの RPC プラグインの有効化についてのトピックを参照してください。

<https://confluence.atlassian.com/display/ALLDOC/Atlassian+Documentation>

1. JIRA\_INSTALL\_DIR/atlassian-jira/WEB-INF/lib ディレクトリにある `atlassian-jira-rpc-plugin.jar` の名前を `atlassian-jira-rpc-plugin.jar.orig.bak` に変更して、既存のプラグインをバックアップします。
2. DTP に同梱された拡張された jar ファイルで、元のファイルを置き換えます。  
[DTP\_HOME]\grs\extras\jira ディレクトリの下に、JIRA のバージョンごとに jar ファイルが置かれています。

バージョン	Jar
3.6.*	[DTP_HOME]\grs\extras\jira\rpc-plugin.jar
3.7.3	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-3.12.1-1.jar
3.10	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-3.10.2-1.jar
3.12	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-3.12.1-1.jar
3.13	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-3.13.1-1.jar
4.1.2	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-4.1.2.jar
4.3	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-4.3.jar
4.4	[DTP_HOME]\grs\extras\jira\atlassian-jira-rpc-plugin-4.4.jar

JIRA 5.x との統合の詳細については 276 ページの「BTS Scanner と RMS Scanner の設定」を参照してください。

3. JIRA サーバーを再起動します。

## DTP に同梱の拡張 JIRA SOA

DTP に同梱された JIRA SOA インターフェイスには、欠陥の履歴を引き出すメソッドが追加されています。このメソッドは [DTP\_HOME]\grs\extras\jira ディレクトリにあります。

このメソッドの詳細については JIRA の Agile Board に掲載されています。

<http://jira.atlassian.com/browse/JRA-10333>

JIRA\_INSTALL\_DIR/atlassian-jira/WEB-INF/lib/atlassian-jira-rpc-plugin.jar には、JIRA への SOA アクセスの実装が含まれています。追加された SOA メソッドについて JIRA の管理者に通知することを推奨します。

DTP は他の JIRA SOA メソッドを使用します。オリジナルの JIRA SOA メソッドは変更されません。そのため、jar ファイルを置き換えても SOA インターフェイスが拡張されるだけであり、JIRA でデグレードが起こることはありません。

この拡張は通常の JIRA ディストリビューションには含まれていませんが、Atlassian の初期承認を受けています。

## DTP 側での JIRA の設定

1. JIRA のスキャンを設定するには、[DTP\_HOME]\grs\config\bts\Jira.xml ファイルで以下の全般的な設定を行います。

設定	説明
<bts>/<name>	JIRA サーバーを識別するためにユーザーが見て分かる名前を指定します (たとえば My JIRA Server)。JIRA サーバーの各インスタンスに固有の名前でなければなりません。この名前は Project Center の [ 欠陥 / 改善検索 ] ページで [ 欠陥トラッキング システム ] ドロップダウン リストの選択肢として表示されます。
<bts>/<url-prefix>	Report Center から JIRA サーバーのバグ詳細へのリンクを作成するために必要な接頭辞。
<bts>/<connection-settings>/<soap-service>	JIRA SOAP サービスのアドレス。通常は次の書式です。 http://<YOUR_JIRA_SERVER_ADDRESS>:<SERVER_PORT>/rpc/soap/jirasoapservice-v2</soap-service>
<bts>/<connection-settings>/<pass>	JIRA ユーザー パスワード。
<bts>/<connection-settings>/<timeout>	BTS Scanner/JIRA の接続タイムアウトをミリ秒単位で指定します。指定されていない場合、Axis のデフォルトのタイムアウトが使用されます。

例 :

```
<connection-settings>
  <soap-service>http://localhost:8080/rpc/soap/jirasoapservice-v2
</soap-service>
  <user>root</user>
  <pass>root</pass>
  <timeout>720000</timeout>
</connection-settings>
```

2. 「解決済みステータス」および「非アクティブの解決」の値を指定します。この情報がないと、Report Center はバグ履歴レポートを作成できません。

例 :

```
<resolved-status>
  <status>Resolved</status>
  <status>Closed</status>
</resolved-status>
```

```
<inactive-resolution>
  <resolution>Duplicate</resolution>
  <resolution>Won't Fix</resolution>
</inactive-resolution>
```

3. (オプション) JIRA の課題をスキャンするためのスコープを指定します。読み込むことができるのは、「JIRA のすべての課題」、「プロジェクトで制限された課題 (JIRA プロジェクトキーで指定)」、および「ユーザー定義の JIRA フィルターで定義された課題 (JIRA フィルター ID で指定)」です。課題を読み込むときにフィルターのためにページ分割を利用できません。たとえば、JIRA へのリクエスト (SOAP 呼び出し) のたびに 1000 件の課題を読み込むようにスキャナーをセットアップできます。1 つの JIRA リクエストにおいて、スキャン対象の項目の数が多すぎる場合 (たとえば 10000 個以上の場合)、ページ分割を利用してメモリの問題を避けることができます。

例:

```
<scanner-scope>
  <!--
    Scan Jira items which belong to the following Jira filters
  -->
  <jira-filters enable-paging="true" items-per-page="1000">
    <filter-id>10035</filter-id>
    <filter-id>10036</filter-id>
  </jira-filters>

  <!--
    Scan the following Jira projects (project keys)
    If <jira-filters> is specified, this <jira-projects> section is skipped
  -->
  <jira-projects>
    <project-key>PROJECTONE</project-key>
    <project-key>PROJECTTWO</project-key>
  </jira-projects>
</scanner-scope>
```

- `enable-paging`  
読み込みのページ分割が有効 (`true`) か無効 (`false`) かを表します。
- `items-per-page`  
一度に何件の課題を読み込むかを指定します (1000 を指定すると、JIRA リクエストごとに 1000 件の課題が読み込まれます)。
- `<filter-id>`  
JIRA で作成されたフィルターの ID を指定します。JIRA で作成されるフィルター定義は、BTS Scanner 構成を JIRA に接続するユーザーが参照できるべきです。JIRA でこのフィルターが作成された後、ユーザーは JIRA からフィルター ID を取得できます。まず、JIRA の Web ページにアクセスし、[Issues] > [Manage Filters] を選択します。[Manage Filters] ページで、フィルターの [edit] リンクをクリックします。次に、このページの URL を見て `filterId` パラメーターを確認します。たとえば次の URL の場合、`filterId` は 10032 です。

```
http://xen3.parasoft.com.pl:8080/secure/
EditFilter!default.jspa?atl_token=a08P-
8w_fk&filterId=10032&returnUrl=ManageFilters.jspa
```

まだそうでない場合、JIRA の他のユーザーとフィルターを共有してください。

#### 4. (オプション) JIRA のカスタム タイプ マッピングを指定します。

デフォルトでは、JIRA の項目をインポートするとき、DTP は JIRA の "New Feature" および "Improvement" タイプを DTP の「フィーチャー リクエスト」として、そして JIRA の "bug" タイプを DTP の「バグ」として扱います。これは XML の以下の部分であらかじめ定義されています。

```
<imported-issue-type>
  <issue-type translate-to="BUG">Bug</issue-type>
  <issue-type translate-to="FEATURE_REQUEST">New Feature</issue-type>
</imported-issue-type>
```

- JIRA でカスタム タイプを定義している場合、このノードを使ってカスタム タイプをインポートし、DTP のバグまたはフィーチャー リクエストにマッピングできます。たとえば、"My Defect" として JIRA のカスタム タイプを定義している場合、以下のように DTP バグにマッピングできます。

```
  <imported-issue-type>
    <issue-type translate-to="BUG">My Defect</issue-type>
    <issue-type translate-to="BUG">Bug</issue-type>
    <issue-type translate-to="FEATURE_REQUEST">New Feature</issue-type>
    <issue-type translate-to="FEATURE_REQUEST">Improvement</issue-type>
  </imported-issue-type>
```

- config xml 中に <imported-issue-type> ノードがない場合、下位互換性のためにデフォルトの構成が使用されます。
- <imported-issue-type> ノードが空の場合、JIRA の項目はインポートされません。

---

## HTTPS/SSL 経由で BTS Scanner と JIRA を連携するための構成

HTTPS/SSL 経由で BTS Scanner と JIRA を連携するには、通常と同じように JIRA 用に BTS Scanner を構成した後で、以下で説明する手順を実行します。

- BTS Scanner 構成を変更する
- キーストア パラメーターを指定して DTP サービスを実行する

---

### BTS Scanner 構成を変更する

SOAP サービス アドレスに以下の変更を加えます。

1. http を https に変更します。
2. 適切な SSL ポート (通常は 443 または 8443) を指定します。



例：

```
<connection-settings>
  <soap-service>https://jira.somecompany.com:8443/rpc/
soap/jirasoapservice-v2</soap-service>
  <user>root</user>
  <pass>root</pass>
</connection-settings>
```

## キーストア パラメーターを指定して DTP サービスを実行する

1. JIRA tomcat の使用する SSL キー ファイルを取得します。

必要に応じて、この手順について以下で詳細を参照できます。

[http://confluence.atlassian.com/display/JIRA/Running+JIRA+over+SSL+or+HTTPS+\(JIRA+ドキュメント\)](http://confluence.atlassian.com/display/JIRA/Running+JIRA+over+SSL+or+HTTPS+(JIRA+ドキュメント))

および

<http://java.sun.com/javase/6/docs/technotes/guides/security/jsse/JSSERefGuide.html#CreateKeystore>

2. キー ファイルを DTP ホストの任意のディレクトリに保存します。
3. DTP Server に `Java -Djavax.net.ssl.trustStore=<TRUSTED KEYSTORE PATH>` パラメーターを設定し、SSL キー ファイルの場所を指定します。

次の例では、キー ファイル名が `.keystore` であると仮定しています。

Linux の場合、たとえばキー ファイルを次の場所に保存したと仮定します。

```
/home/ser/.keystore
```

1. `DTP_HOME/bin/reportserver.sh` を編集します。

2. 変更前：

```
export CATALINA_OPTS="-D$PCC_RECOGNITION \
```

変更後：

```
export CATALINA_OPTS="-D$PCC_RECOGNITION -Djavax.net.ssl.trustStore=/
home/ser/.keystore \
```

Windows の場合、たとえばキー ファイルを次の場所に保存したと仮定します。

```
D:\home\user\dtp\.keystore
```

1. `regedit` コマンドを実行して Windows レジストリ エディターを開きます。
2. Parasoft DTP サービスの呼び出しパラメーターを探します。

通常、次の場所にあります。

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Parasoft DTP
Server Service
```

3. 新しい JVM Option Number パラメーターを追加します。

たとえば、番号が 0 から 14 までの 15 個のパラメーターがある場合、次のように 16 個目のパラメーターを追加します。

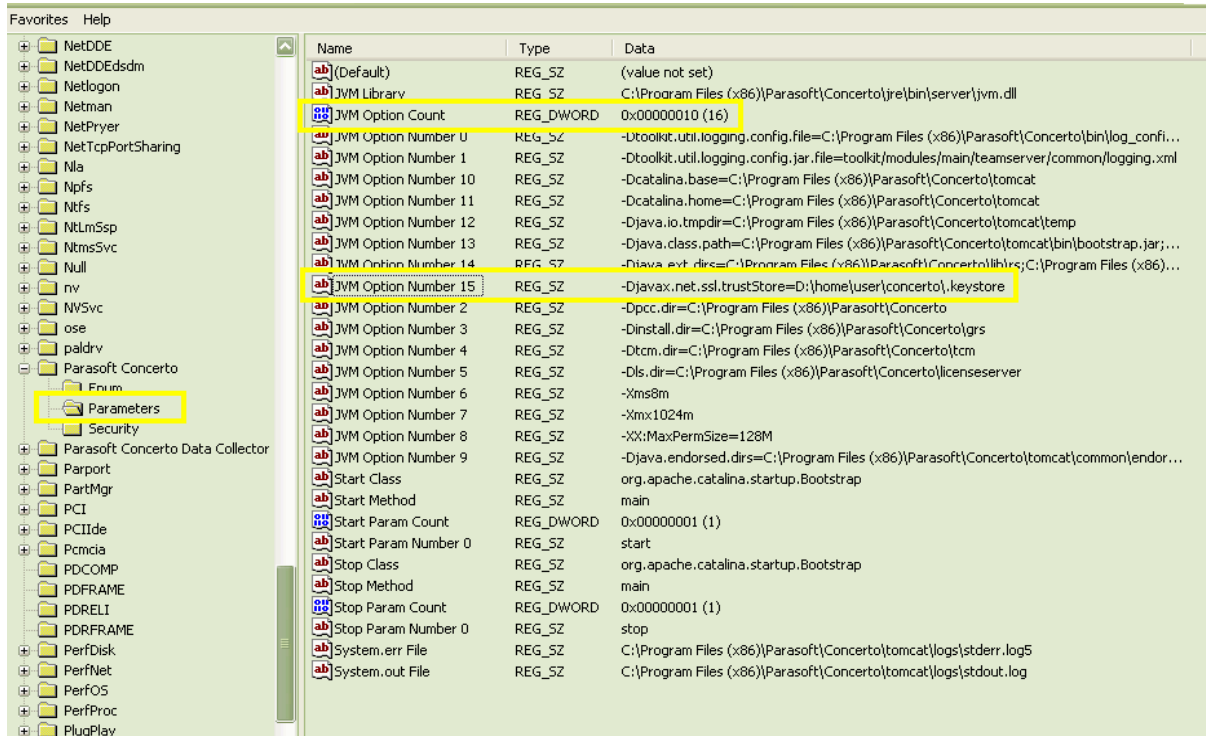
JVM Option Number 15

値：

-Djavax.net.ssl.trustStore=D:\home\user\ntp\keystore

4. パラメーターのカウンター JVM Option Count の値を増やします。たとえば、JVM Option Count の値が 15 だった場合、16 にします。

### Regedit での JVM Option Count の編集



## HTTP ベーシック認証を使用した BTS Scanner と JIRA の連携

HTTP ベーシック認証を使用して BTS Scanner と JIRA を連携するには、通常どおり JIRA 用に BTS Scanner を構成した後で、以下の手順を実行します。

`${DTP_HOME}\grs\config\bts\Jira.xml` ファイルの接続設定に `<basic-http-authentication>` を追加します。

```
<basic-http-authentication>
  <user>admin</user>
  <pass encrypted="false">admin</pass>
</basic-http-authentication>
```

例：

```
<connection-settings>
  <soap-service>https://jira.somecompany.com:8443/rpc/soap/
jirasoapservice-v2</soap-service>
  <user>root</user>
  <pass>root</pass>
  <basic-http-authentication>
    <user>admin</user>
    <pass encrypted="false">admin</pass>
  </basic-http-authentication>
</connection-settings>
```

**注意：** <basic-http-authentication> タグ内のユーザーおよびパスワードが HTTP ヘッダーのパラメーターとして JIRA サーバーに送信されます。

---

# IBM Rational Change および Rational Synergy との統合

このセクションでは、Rational Change の成果物を Development Testing Platform (DTP) にインポートする方法と DTP にインポートされた Rational Change の成果物を使用するようにタスクアシスタントを設定する方法について説明します (381 ページの「IBM Rational Change および Rational Synergy との統合」)。

---

## Rational Change との統合

Rational Change と Rational Synergy 用の BTS Scanner を使用すると、変更リクエスト (および関連するタスク) を Rational Change から DTP にインポートすることができます。Rational Change の変更リクエストは DTP では要求になります。Rational Change と Rational Synergy は BTS/RMS Scanner によって同期化されます。BTS/RMS Scanner は自動的に実行するか、必要に応じて手動で実行できます。

1. DTP\_HOME/grs/config/bts/examples/ExampleSynergyScannerConfig.xml ファイルのコピーを作成します (このファイルには Synergy の設定のサンプルが含まれます)。実際の Synergy サーバーと DTP Server に合わせて、コピーしたファイルを編集します。個々の設定要素がこの XML ファイル中に記述されています。
2. このファイルを DTP\_HOME/grs/config/bts にコピーします。ここまでの操作が終了すると、指定した Rational Change サーバーが接続され、.xml ファイルで指定された "ccm" クエリーが 15 分ごとに実行されます (または、ユーザーが Project Center の [要求] ページで [再スキャン] ボタンをクリックしたときに実行されます)。

クエリーによって発見された変更要求 (およびその変更要求に関連付けられた子要求とタスク) は、DTP のターゲット プロジェクト中の要求およびタスクと同期化されます。

Rational Change で行われる変更は DTP で次のように更新されます。

- 特定の要求またはタスクが DTP に存在しない場合、その要求またはタスクは追加されます。
- 特定の要求またはタスクが Rational Change で変更された場合、その要求またはタスクは DTP で更新されます。Rational Change の変更リクエスト ID とタスク ID が DTP の要求のオリジナル ID とタスクのオリジナル ID フィールドにそれぞれ格納されます。

**注意**

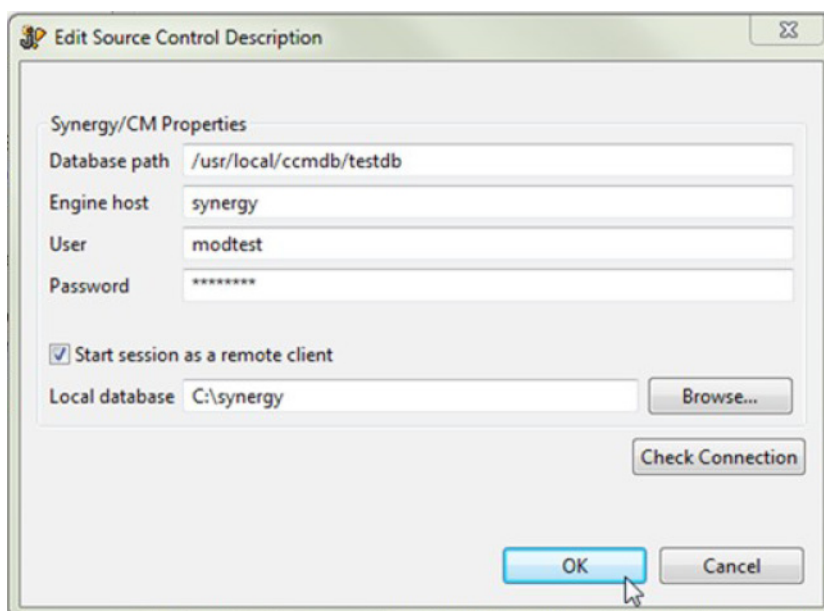
- .xml 設定ファイルで設定された Synergy の "ccm" クエリーは、DTP ターゲット プロジェクトに緊密に統合されます。たとえば、特定の構成が返す変更リクエストを減らすために、クエリーを変更または削除した場合、DTP ターゲット プロジェクト中の未解決の要求が削除されます。削除された要求に関連付けられたすべてのタスクの種類が "スタンダードアロン" に変わります。タスクが削除されることはありません。
- DTP の要求オリジナル ID およびターゲット プロジェクトは、DTP/Rational Change で対応する項目を探すときに考慮されます。たとえば、ある要求がインポートされ、その要求と同じオリジナル ID を持つ要求が DTP ターゲット プロジェクトに存在しない場合、DTP はその要求を作成します。インポートされた要求と同じオリジナル ID を持つ要求が DTP ターゲット プロジェクトに存在する場合、その要求は更新されます。
- 場合によっては、DTP プロジェクトと同期化するように "ccm" クエリーを定義することもあります。そして、構成の DTP ターゲット プロジェクト名は変更しつつ、クエリーに関しては、別の DTP プロジェクトと同じ変更リクエストをインポートできるよう、そのままにすることもあります。その後、Rational Change の「まだ DTP にインポートされていない変更リクエスト」または「前回 DTP にインポートされてから修正があった変更リクエスト」を、"ccm" クエリーで検索して DTP にインポートする場合、オリジナル ID が同じ要求が DTP にすでに存在するが、その要求が別のプロジェクト中にあると、以下の処理が実行されます。
  - 要求はオリジナル プロジェクトから削除されます。
  - 削除される要求に関連付けられたすべてのタスクについて、その種類が "スタンダードアロン" に変わります。タスクは削除されません。
  - 要求は新しいプロジェクトにインポートされます。
  - このインポートされる要求にタスクが関連付けられている場合、オリジナル ID に基づいて、タスクは新規に作成されるか、またはこの要求に移動されます。

3. クエリーが検索する変更リクエストごとに、関連する子要求とタスクも同期化されます。たとえば、新しいタスクが Rational Change 中の変更リクエストに追加される場合、このタスクは同期化の際に DTP にも追加されます。

この時点で、DTP ターゲット プロジェクトで要求とタスクが見えるはずですが、DTP にインポートされた要求とタスクのオリジナル ID フィールドに、Rational Change での ID が表示されている点に注意してください。

## DTP および Rational Change Synergy と連携するための Parasoft Test タスク アシスタントの設定

1. Synergy からデータをインポートした DTP プロジェクトを構成します。次の操作を行います。
  - a. Parasoft Test で、Synergy ソース リポジトリにアクセスするためのテキスト プロパティを取得します。それには、Parasoft Test で Synergy のソース管理を定義し、その設定をファイルにエクスポートします。



エクスポートしたプロパティを DTP プロジェクトの [Parasoft Test 設定] タブに貼り付けます。以下はエクスポートしたプロパティの例です。

```
scontrol.rep1.synergy.dbpath=/usr/local/ccmdb/testdb
scontrol.rep1.synergy.host=synergy
scontrol.rep1.synergy.local_dbpath=C:\\tmp\\local_synergy_db
scontrol.rep1.synergy.login=modtest
scontrol.rep1.synergy.password=6e6b703e6952746f
scontrol.rep1.synergy.remote_client=true
scontrol.rep1.type=synergy
scontrol.synergy.exec=ccm
```

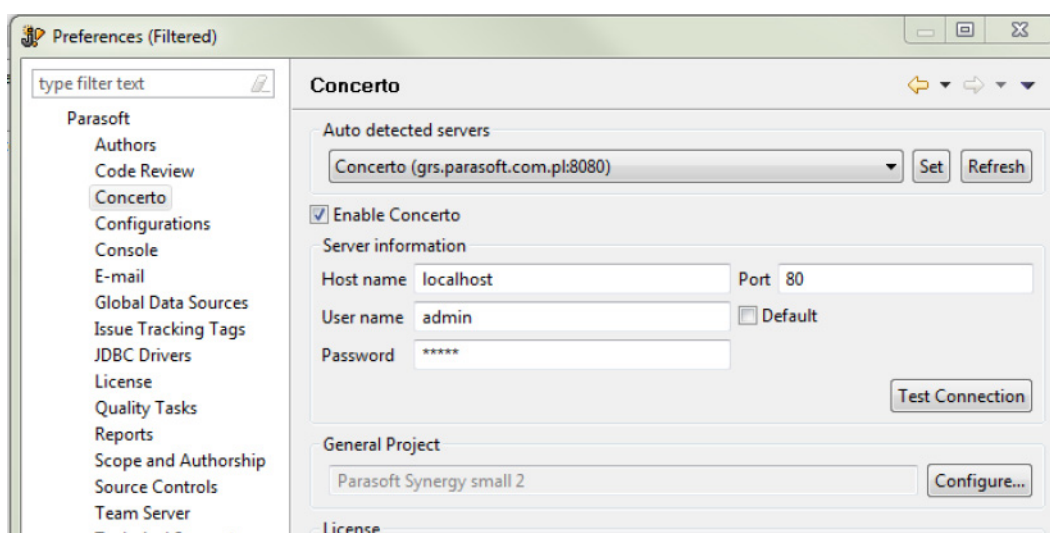
この設定は、指定の DTP プロジェクトに Parasoft Test が接続するときに使用されます。この設定はプロジェクト固有です。

- b. DTP の管理ページで、作業するプロジェクトを開いて編集し、[Parasoft Test 設定] タブをクリックします。上記のプロパティを貼り付けます。この設定をすべての DTP プロジェクトに対してグローバルに指定する場合は、プロジェクトの [Parasoft Test 設定] タブではなく、[Administration] > [Settings] > [Parasoft Test] で表示されるページでこの情報を貼り付けます。
- c. [Administration] > [Settings] > [Parasoft Test] ページに移動し、次の行を [Global Parasoft Test Settings] フィールドに貼り付けます。

```
tasksImportedMode=true
```

この設定は、この DTP Server に接続するタスク アシスタントを Synergy モードに切り替えます。つまり、タスク アシスタントはタスク ID であるかのようにタスク オリジナル ID を処理します (Synergy のタスク ID の値は、タスク オリジナル ID として DTP で表示されます)。

2. DTP Server ホストで、Synergy クライアントをインストールします。ccm コマンドを使って、DTP は Synergy のタスクと関連付けられたファイルを読み込みます (DTP は、タスク アシスタントが照会するときにこれらのファイルを読み込みます)。
3. 以下のことを確認してください。
  - DTP Server サービスから見えるシステム パス上に "ccm" コマンドがある。
  - たとえスペースを含む場合でも、定義されたパスがダブルクォートで囲まれていない。
4. Parasoft Test で DTP の設定を開き、DTP Server を指定します。そして [General Project] に、Synergy の要求とタスクをインポートした、構成したプロジェクトを指定します。



ここまでの操作によって、タスク アシスタントと Synergy を連携する準備ができました。つまり、DTP からタスクを検索するクエリーをユーザーは定義できます (そして DTP は Synergy からタスクを検索します)。特定のタスクに関連付けられたファイルが Synergy にある場合、タスク アシスタントでそのタスクをアクティブにすると、DTP はそれらのファイルのリストを Synergy から読み込んで タスク アシスタントに表示します。

## トラブルシューティング

### ccm プログラムを実行できない

Synergy タスクをアクティブにしようとする、Parasoft Test のタスク アシスタントで「Fail reason: Cannot run program ""ccm"": CreateProcess error=2」というエラーが出る。

#### 考えられる原因：

- Synergy クライアント (ccm コマンドを含む) が DTP Server マシンにインストールされていない。
- DTP サービスに見えるシステム パス上に ccm コマンドがない。

- ccm.exe コマンドへのパスが Windows で定義されているが、二重引用符で囲まれて定義されている。

**解決方法：** Synergy クライアント (ccm コマンドを含む) が DTP Server マシンで使用できること、DTP サービスに見えるシステムパス上に ccm コマンドがあること、および Windows の ccm コマンドへのパスが (たとえパスにスペースが含まれる場合でも) 引用符で囲まれていないことを確認します。

### 新しい Synergy/CM セッションを開始できない

Synergy タスクをアクティブにし、関連するファイルを見るために [タスク アシスタント] ビューに移動すると、「Cannot start new Synergy/CM session. Warning: IBM Rational Synergy startup failed」というエラーが出る。

**考えられる原因：** DTP Server から ccm コマンドは見えているが、DTP サービスを実行しているシステムユーザーに Synergy セッションを開始する権限がない。

**解決方法：** Synergy セッションを開始する権限を持った Windows システムユーザーとして DTP サービスを開始します。Windows で DTP サービスを実行するユーザーを変更する方法については 896 ページの「ソース表示機能」の「解決方法 2」を参照してください。

### デバッグ モードを有効にする

次の a) b) c) に該当する場合、DTP Server ログをデバッグ レベルに切り替えます。操作手順については 899 ページの「デバッグ ログ モードへの切り替え」を参照してください。

- a) Synergy の特定のタスクに関連付けられたファイルがあるが、それらのファイルをタスク アシスタントで見ることができない。
- b) それらのファイルを照会すると、タスク アシスタントでエラーが発生する。
- c) 上記のヒントが役に立たなかった。



## コードレビューの統合

Development Testing Platform (DTP) は、Parasoft Test (SOAtest、Jtest など) を介してユーザーがセットアップしたコードレビューと統合します。その過程で Parasoft Test はコードレビューエンジンの役割を果たし、チームメンバーは Parasoft Test を介してコードレビュータスクを処理します。

DTP は、コードレビューの結果を格納するために使用され、コードレビューの結果についての洞察を提供します。

Parasoft Test でコードレビューを設定して実行する方法については『Parasoft Test ユーザーズガイド』を参照してください。

## プロジェクトのコードレビューレポートの設定

コードレビューレポートは、選択されたプロジェクトについてコードレビューの統計を示します (コードレビューレポートについては 68 ページの「コードレビューレポート」を参照)。

コードレビューレポートを生成するときに DTP は次のことを考慮します。

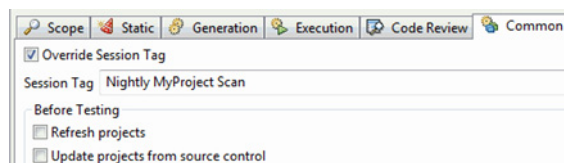
- そのプロジェクトのメンバーに関連付けられた結果はどれか。
- コードレビューセッションの識別子 / セッションタグでマークされている結果はどれか。

レポートを設定するには、次の操作を行います。

1. プロジェクトのチームメンバーシップのリストに、コードレビュープロセスに関わるユーザーが含まれていることを確認します。
  - チームメンバーの構成の詳細については 230 ページの「プロジェクトの作成と構成」を参照してください。
2. プロジェクトのコードレビューフィルターに、Parasoft Test Code Review テストコンフィギュレーションに一致するセッションタグセットがあることを確認します。
  - コードレビューフィルターの詳細については 236 ページの「コードレビューフィルター」を参照してください。

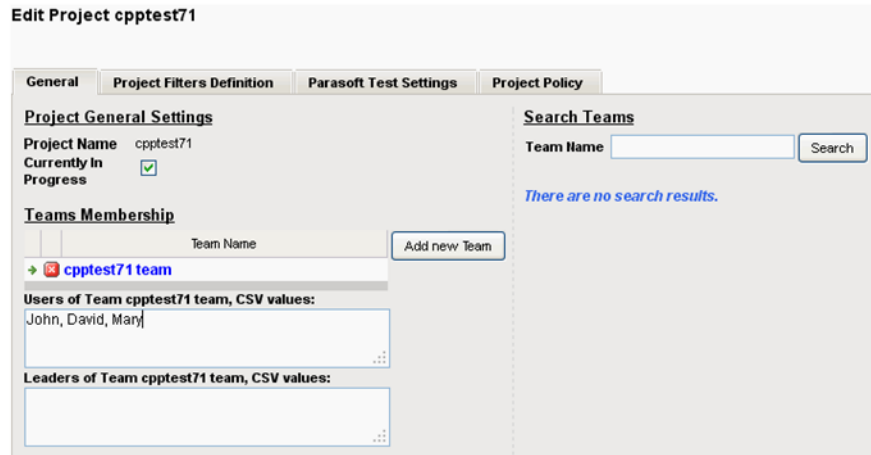
たとえば、次の状況を想定してみましょう。

- John、David、および Mary のコードレビューの結果を参照したい。
- チームの Code Review テストコンフィギュレーションが Nightly MyProject Scan セッションタグを使用している。

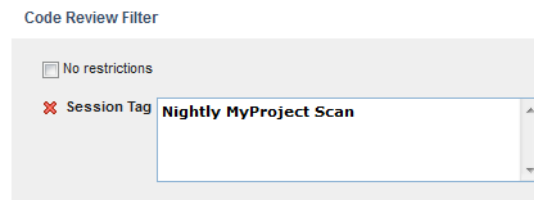


この場合、以下の設定を DTP で行います。

- チームメンバーシップのリストに John、David、Mary を追加します。



- Nightly MyProject Scan セッション タグを使うようにコードレビューフィルターを設定します。



## コードレビューデータの格納

DTP でコードレビューデータを格納する方法は 2 つあります。

- Report Center データベースを使用する（推奨。より効率的にデータを格納できます）。
- Team Server を使用する（下位互換のためにサポートされている）。

コードレビューデータの格納方法によって、DTP の統合手順は異なります。

### Report Center データベースを使用する場合

このオプションを推奨します。このオプションを選択した場合、Parasoft Test で行われたコードレビューの構成で十分です。DTP 側でさらに構成を行う必要はありません。

Report Center のレポートは、Report Center データベースからデータを取得するようにデフォルトで構成されます。

## Team Server を使用する場合

Report Center のレポートは、デフォルトでは Report Center データベースからデータを取得するように設定されています。Team Server からデータを取得するように切り替えるには、次の操作を行います。

1. Report Center で、[ 管理 ] ドロップダウンメニューから [Report Center] を選択し、[ 設定 ] > [Report Center] > [ サーバー ] をクリックします。[ 非推奨の Team Server ベースのコード レビューを使用 ] チェックボックスをオンにします。
2. DTP\_HOME/grs/config/CRHistoryScanners.xml ファイルを編集し、Team Server からデータを読み込んで Report Center データベースに保存するようにナイトリー ジョブを設定します。コード レビュー データを格納した Team Server の各サーバーについての情報を提供します。以下はその例です。

```
<configuration>
  <cr-history-scanner>
    <teamserver>
      <host>teamserver1.company.com</host>
      <port>1111</port>
      <login>login</login>
      <password>pass</password>
      <timeout>7200000</timeout>
    </teamserver>
  </cr-history-scanner-->
  <cr-history-scanner>
    <teamserver>
      <host>teamserver2.company.com</host>
      <port>1111</port>
      <login>login2</login>
      <password>pass2</password>
      <timeout>7200000</timeout>
    </teamserver>
  </teamserver>
</cr-history-scanner>
  ...
</configuration>
```

3. DTP Server を再起動して変更を反映します。

Report Center Code Review History Scanner ジョブが起動すると、<cr-history-scanner/> セクションに指定されたすべての Team Server が順番にスキャンされます。そしてデータが Report Center データベースに保存されます。

スキャン処理には時間がかかるため、このジョブはあらかじめ構成されているとおり、独立したグループとして実行するべきです。time-out パラメーターには、スキャン処理が完了するまでのタイムアウト時間をミリ秒単位で指定します。デフォルト値の 7,200,000 ミリ秒を設定するべきです。

**警告！** Team Server リポジトリの初回のスキャンは非常に長い時間がかかる場合があります。これは、Report Center Code Review プラグインがインデックスをゼロから作成する必要があるからです。コード レビュー アクティビティの完全な履歴を取得するには、スキャン セッションを何度か実行する必要がある場合もあります。その後のスキャンにかかる時間は一日一回実行できる程度に短縮されるでしょう。

# Report Center と Emma の統合

Report Center は、Java のコード カバレッジ ツールである Emma と統合できます。Emma の機能テスト カバレッジの結果が Report Center レポートに表示されます。390 ページの「Emma の結果の参照」を参照してください。

1. Report Center で、[ 管理 ] ドロップダウン メニューから [Report Center] を選択します。
2. [ 設定 ] メニューの [ 統合 ] > [Emma] をクリックします。
3. すべてのテスト対象クラスのカバレッジを含む Emma の .xml ファイルの URL を入力します。

Report Center は、この URL の場所を定期的にスキャンします。新しい結果が発見された場合、その結果は Report Center データベースにロードされます。

**注意：** 指定した URL にある .xml ファイルは HTTP サーバーによって提供されていなければなりません。 .xml 出力のサンプルを Emma の公式 Web サイトで参照できます。

[http://emma.sourceforge.net/coverage\\_sample\\_c/coverage.txt](http://emma.sourceforge.net/coverage_sample_c/coverage.txt)

4. ( オプション ) すべてのテスト対象クラスのカバレッジを含む Emma の .html ファイルの URL を入力します。この HTML ファイルには、ファイルには Emma の最終実行時の結果が含まれていなければなりません。

UTL の場所は定期的にスキャンされません。DTP は、Report Center のカバレッジ レポートの [Emma Coverage] 列から Emma のカバレッジ レポートにリンクします。390 ページの「Emma の結果の参照」を参照してください。

指定した URL にある .html ファイルは HTTP サーバーによって提供されていなければなりません。Emma HTML レポートのサンプルを Emma の公式 Web サイトで参照できます。

[http://emma.sourceforge.net/coverage\\_sample\\_a/index.html](http://emma.sourceforge.net/coverage_sample_a/index.html)

5. [User Attribute: Project] フィールドで Test Group Property 属性フィルターを指定します。このフィルターは、どの DTP プロジェクトを Emma の結果と関連付けるかを指定します。フィルターの値は Edit Project ページで設定されます。詳細については 234 ページの「テスト グループ プロパティ フィルター」を参照してください。

PARASOFT Development Testing Platform Report Center

Main Projects Tools Settings Reports Help

Emma Integration Settings

The below Emma results file URIs can point to remote file (<http://example.com/coverage.xml>) or local file on DTP server (<file:///c:/emma/coverage.xml>)

Emma Scanner Configuration1

Emma results .xml file URI:

Emma results .html file URI:

User Attribute: Project

[Add Another Configuration]

Save

6. さらに設定を行うか、[Save] をクリックして操作を続けます。
7. DTP\_HOME\grs\config\CronConfig.xml ファイルを編集し、次の xml ノードに移動します。

```

<Job allowFromHour="0" allowToHour="23" frequency="1440"
id="Emma Scanner"runDayOfWeek="*" runHour="2" runMinute="0">
    <class allowInSlave="false"
name="com.parasoft.grs.rserver.cronjobs.EmmaScannerJob"
priority="1"/>
</Job>

```

このノードには、ステップ 1 で定義された場所をスキャンする周期的な Report Center バックグラウンド ジョブが設定されています。この例では毎晩午前 2 時にジョブが実行されます。

必要に応じて実行時間を変更します。変更を行ったら、DTP Server を再起動して変更を有効にします。

#### 8. Report Center のカバレッジ レポートを Emma モードに切り替えます。

##### a. 以下のファイルを編集します。

- DTP\_HOME/grs/xreports/architect\_dashboard/CoverageDetails.xml
- DTP\_HOME/grs/xreports/architect\_dashboard/CoverageOverview.xml
- DTP\_HOME/grs/xreports/architect\_dashboard/CoverageOverviewDetails.xml
- DTP\_HOME/grs/xreports/architect\_dashboard/composite\_desc.xml
- DTP\_HOME/grs/xreports/lite\_dashboard/composite\_desc.xml

##### b. 次の XML 要素に移動します。

- `<parameter name="show_emma_results">false</parameter>`

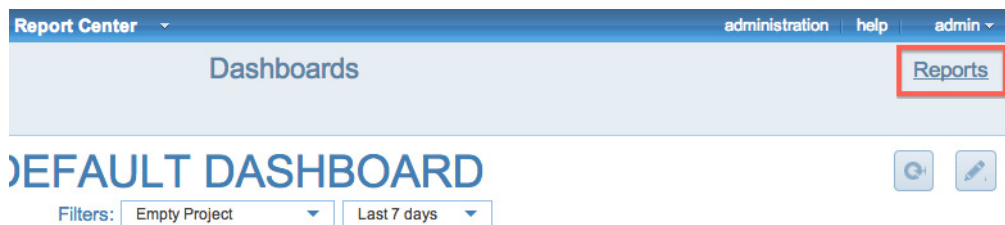
##### c. 値を true に設定します。

- `<parameter name="show_emma_results">true</parameter>`

Report Center と Emma の統合が完了します。

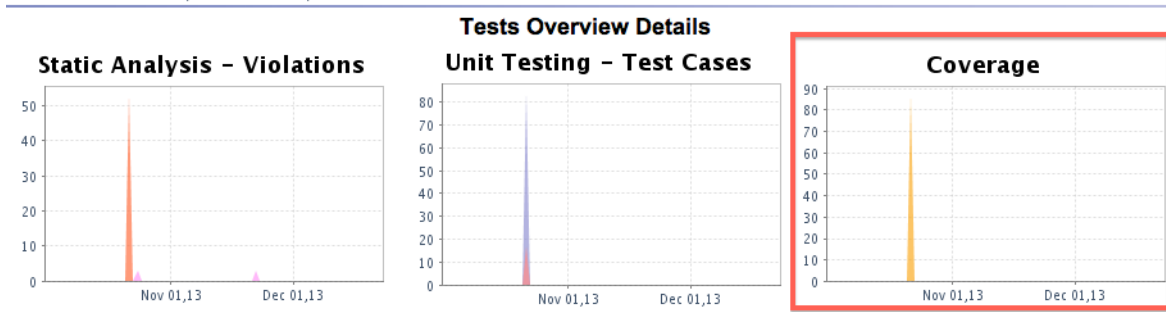
## Emma の結果の参照

#### 1. Development Testing Platform レポート ビューを開きます。



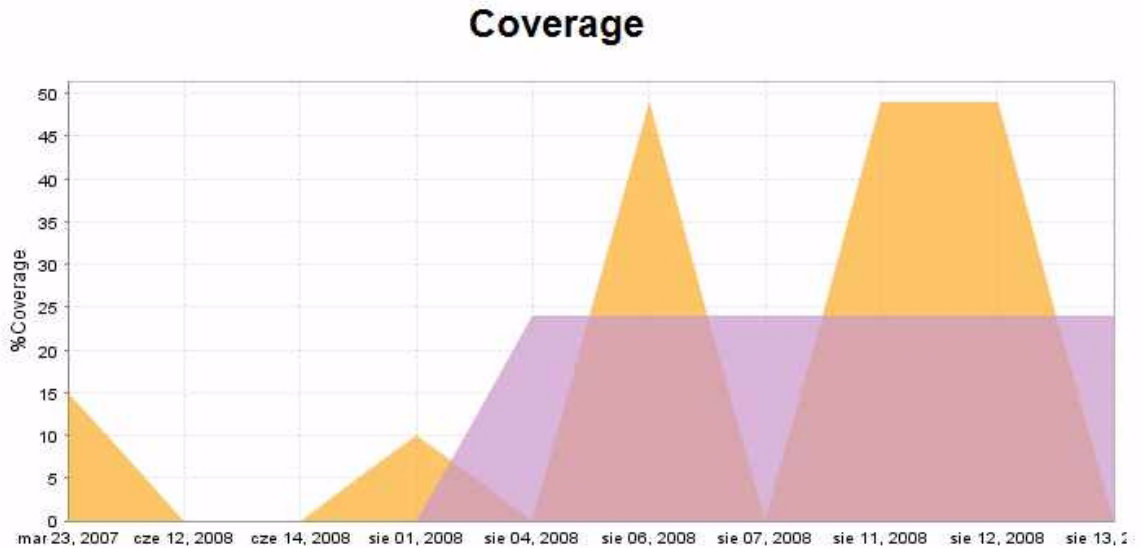
#### 2. [ テスト ] > [ テストの概要 ] をクリックします。

#### 3. カバレッジ レポートをクリックします。



Emma モードでは、カバレッジ レポートとドリル ダウン レポートに [Emma Coverage] 列が表示され、Emma からレポートされたカバレッジが表示されます。

列のヘッダーをクリックして Emma HTML ファイルにアクセスします。



Date	Unit Tests Coverage	Emma Coverage
sie 13, 2008	0% (0 / 0)	24% (5466 / 23157)
sie 12, 2008	49% (1582 / 3256)	24% (5466 / 23157)
sie 11, 2008	49% (791 / 1628)	24% (5466 / 23157)
sie 07, 2008	0% (0 / 0)	24% (5466 / 23157)
sie 06, 2008	49% (791 / 1628)	24% (9110 / 38595)
sie 04, 2008	0% (0 / 0)	24% (1822 / 7719)
sie 01, 2008	10% (791 / 8035)	0% (0 / 0)
cze 14, 2008	0% (0 / 0)	0% (713 / 156813)
cze 12, 2008	0% (0 / 0)	0% (1426 / 313626)
mar 23, 2007	15% (26250 / 175526)	0% (0 / 0)

---

# Report Center とソース管理システムの統合

Report Center とソース管理システム (SCM) を統合すると、コミットされたソース ファイルからのデータを基に Report Center で広範な統計を表示することができます。

DTP を SCM と統合すると、ソース リポジトリに保存されたファイルを SourceScanner (SDLC Extensions に同梱) がスキャンし、Report Center データベースにリビジョン情報を格納します。Report Center が収集するデータを使用して正確なレポートが生成されます。Report Center は、ソース コードに関連するレポートに、適切なリポジトリのソース コードを表示できます。

DTP を SCM に統合しない場合は、DTP Engine から DTP Server に送られたソース ファイルの内容を参照することができます。詳細については 644 ページの「DTP Engine がパブリッシュしたソース コードの表示」を参照してください。

## Parasoft SDLC Extensions

SourceScanner は Development Testing Platform には同梱されていません。SourceScanner は Parasoft SDLC Extensions モジュールの一部です。

---

## SourceScanner の構成

SourceScanner を構成して、SourceScanner から Report Center にデータを送信します。

SourceScanner は他の SDLC integration extensions とともに提供されています。

SourceScanner の構成方法については、『Parasoft SDLC Integration Extensions ユーザーズ ガイド』を参照してください。

---

## Report Center プロジェクトに対するソース管理フィルターの定義

SourceScanner から Report Center にデータを送信できるように構成したら、特定のプロジェクトのレポートに表示させたいリポジトリ ファイルを定義する必要があります。詳細な手順については 235 ページの「ソース管理フィルター」を参照してください。

**重要!** ソース リポジトリの特定の場所をスキャンする場合、1 つの SourceScanner プロジェクトだけを使用することを強く推奨します。つまり、2 つのブランチおよび同じファイルのトランクをスキャンする場合、3 つの個別の SourceScanner プロジェクトを定義して実行するのではなく、1 つのプロジェクトに定義するべきです。



## ソース管理システムおよびファイルのソースコードの統計の表示

SourceScanner を構成し、さまざまな Report Center プロジェクトに対してソース管理フィルターを定義すると、Report Center のレポート ビューとソース コード ビューにソース コード データに基づいて SCM の統計を表示することができます。詳細については以下のセクションを参照してください。

- 115 ページの「違反エクスプローラー」
- 132 ページの「カバレッジ エクスプローラー」
- 137 ページの「テスト エクスプローラー」
- 153 ページの「レポート」

## ソースコードの参照 (ソース表示機能)

DTP を設定して Report Center に SCM データを含めることができます。DTP がリポジトリのソースを表示できるようにするには、Report Center と同じマシン上にリポジトリ クライアントをインストールする必要があります。次の表は、DTP と統合できる SCM の一覧です。

SCM	タスク
Accurev	Accurev クライアントがインストールされていること、および <code>accurev</code> コマンドがシステムパスに含まれていることを確認します。 問題のトラブルシューティングに関しては 896 ページの「ソース表示機能」を参照してください。 レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。
ClearCase	ClearCase クライアントがインストールされていること、および <code>cleartool</code> コマンドがシステムパスに含まれていることを確認します。 問題のトラブルシューティングに関しては 896 ページの「ソース表示機能」を参照してください。 レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。
CVS	CVS クライアントをインストールする必要はありません。 レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。



SCM	タスク
Jazz (Rational Team Concert)	<p>Jazz 2.0.0.2 がサポートされます。</p> <p>Jazz Plain Java Native Libraries が DTP マシンになければなりません。DTP は SourceScanner を介して Jazz Plain Java Native Libraries を発見することができます。</p> <p>Jazz コマンドライン クライアントは Jazz Eclipse Client に含まれるため、Jazz Eclipse Client が DTP マシンになければなりません。また、Jazz コマンドライン クライアントへのパスを PATH 環境変数に追加する必要があります。このパスは通常 [Jazz Directory]/scmtools/eclipse/ の下です。</p> <p>SourceScanner 5.4.2 が必要です。SourceScanner と DTP は同じマシン上になければなりません。</p> <p>レポート ビュー (153 ページの「レポート」) だけがサポートされます。</p>
GIT	<p>GIT クライアントがインストールされていることを確認します。</p> <p>git コマンドがシステム パスになければなりません。</p> <p>ローカル リポジトリ (Source Scanner で ROOT として設定) および DTP が同じマシン上になければなりません。</p> <p>レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。</p>
Microsoft Team Foundation Server	<p>違反エクスプローラーだけがサポートされます。115 ページの「違反エクスプローラー」を参照してください。</p>
Microsoft Visual SourceSafe	<p>違反エクスプローラーだけがサポートされます。115 ページの「違反エクスプローラー」を参照してください。</p>
Perforce SCM	<p>Perforce クライアントがインストールされていることを確認します。</p> <p>p4 コマンドがシステム パスになければなりません。</p> <p>レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。</p>
Serena Dimensions	<p>DM_ROOT 環境変数が設定されていて正しいことを確認します (たとえば <code>export DM_ROOT=/opt/serena/dimensions/12.2/cm</code>)。</p> <p>LD_LIBRARY_PATH 環境変数に Serena Dimensions libs へのパスが含まれていることを確認します (たとえば <code>export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/opt/serena/dimensions/12.2/cm/lib</code>)。</p> <p>レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。</p>

SCM	タスク
StarTeam	<p>Borland StarTeam SDK が [DTP_HOME]/lib/thirdparty ディレクトリにインストールされていることを確認します。SDK をインストールした後に DTP サービスを再開始する必要があります。</p> <p>Borland StarTeam SDK は Borland の Web サイトから無料でダウンロードできます。</p> <p>レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。</p>
Subversion	<p>Subversion クライアントがインストールされていることを確認します。</p> <p>svn コマンドがシステム パスになければなりません。</p> <p>レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。</p> <p>問題のトラブルシューティングに関しては 896 ページの「ソース表示機能」を参照してください。</p>
Synergy CM	<p>Synergy クライアントがインストールされていること、ccm コマンドがシステム パスに含まれていることを確認します。</p> <p>レポート ビュー (153 ページの「レポート」) および違反エクスプローラー (115 ページの「違反エクスプローラー」) がサポートされます。</p> <p>問題のトラブルシューティングに関しては 896 ページの「ソース表示機能」を参照してください。</p>

---

# PTC Integrity ソース管理コントロール拡張との統合

Parasoft DTP は PTC Integrity (MKS) 10.x と統合します。これは次の機能を可能にします。

- スコーピングと Parasoft Test タスク作成者の特定をサポートする
- Parasoft コード レビューの実行をユーザーに許可する
- Parasoft Test タスクからソース コードを参照する
- コード レビューの自動化プロセスを容易にする

PTC Integrity 10.4 を使って以下の統合サポートがテストされ、確認されています。ただし、この統合は PTC Integrity 10.0 ~ 10.4 をサポートする MKS API 4.10 をサポートするために実装されました。

---

## 統合用パッケージ

Parasoft は、MKS 統合のために次のパッケージを提供しています。

- `xtestMKS.jar`: PTC Integrity 10.x をサポートするための Parasoft ソース管理の実装
- `mksapi.jar`: PTC Integrity Java API

---

## 要件

- PTC Integrity 10.x クライアントをインストールする必要があります。Parasoft は、情報を取得するために、PTC Integrity クライアントの Source Integrity (SI) コマンドラインと Java API の両方を使用します。
- ツールが使用するプロジェクト用のサンドボックスを初めに作成する必要があります。すべてのファイルをこのサンドボックスに置く必要があります。
- Parasoft Test 9.5.10 (Jtest、C++test、SOAtest) 以上を必ずインストールする必要があります。
- Linux にインストールされた Parasoft DTP。Windows サーバー上の DTP については、この統合は現在サポートされていません。

---

## インストールと設定

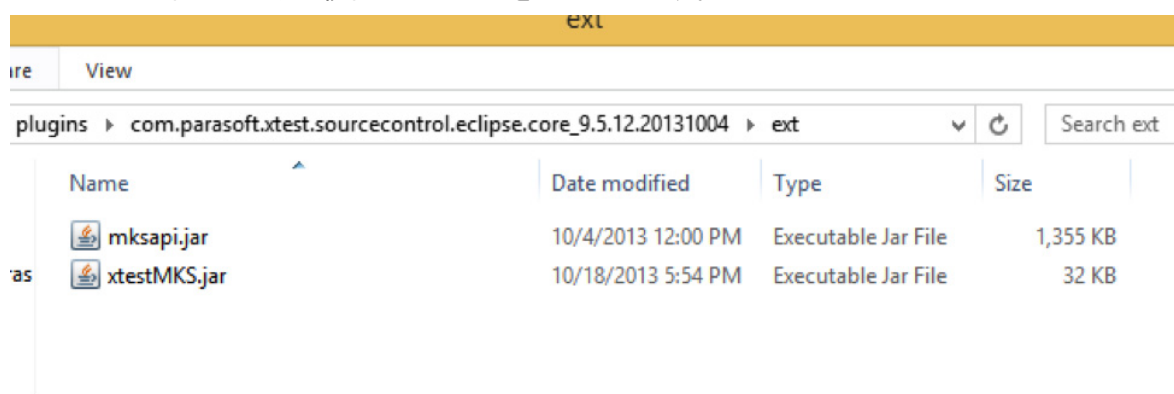
パッケージは Parasoft Test と DTP でインストールされ、設定されます。

---

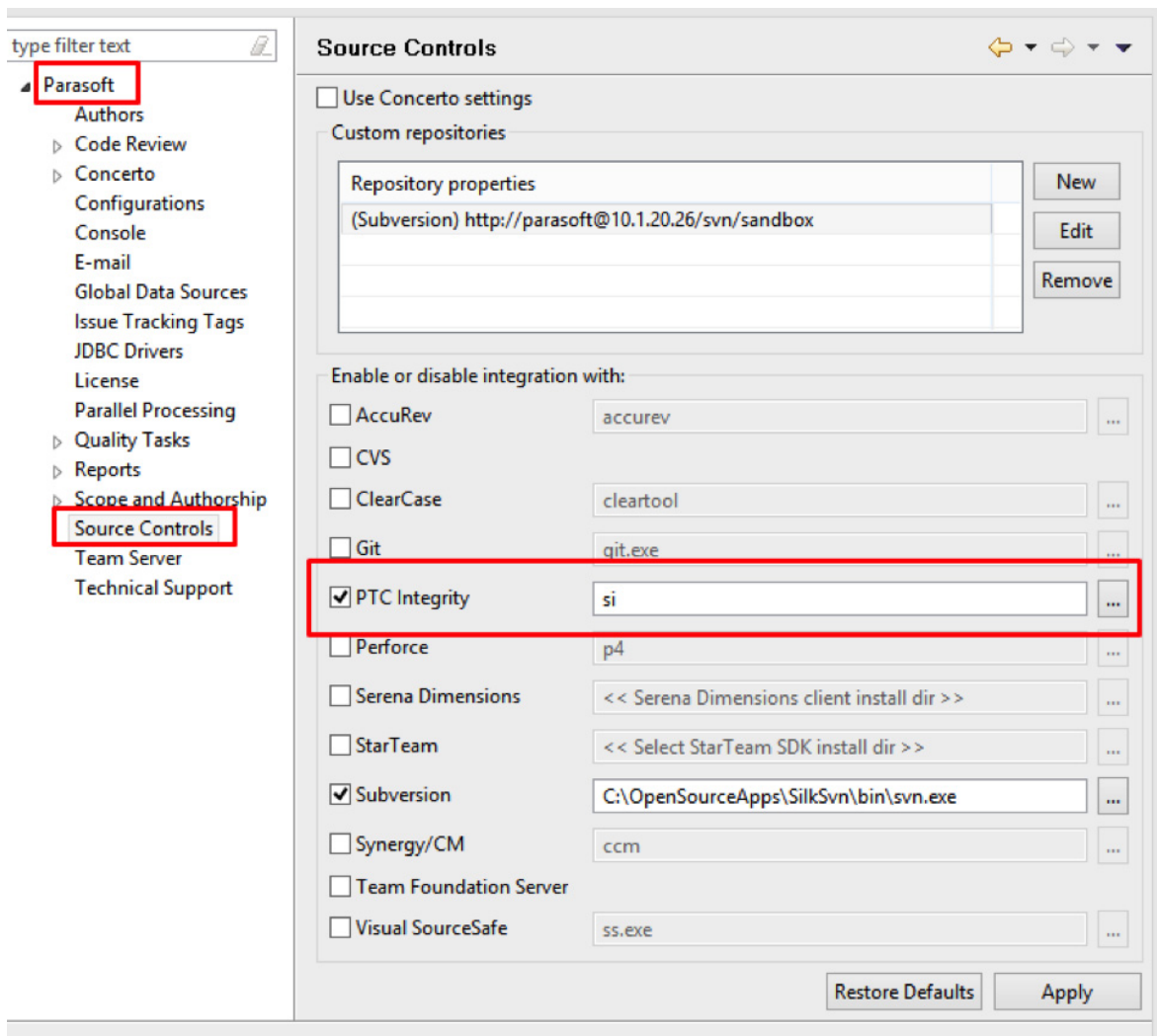
### Parasoft Test へのインストール

1. インストール ディレクトリを開きます。

- Windows 32-bit スタンドアロン : C:\Program Files\Parasoft\Test\9.5
  - Windows 64-bit スタンドアロン : C:\Program Files (x86)\Parasoft\Test\9.5
  - Windows 32-bit プラグイン : C:\Program Files\Parasoft\Test for Eclipse\9.5
  - Windows 64-bit プラグイン : C:\Program Files (x86)\Parasoft\Test for Eclipse\9.5
  - Linux スタンドアロン : \$HOME/parasoft/test/9.5
  - Linux プラグイン : \$HOME/parasoft/test for eclipse/9.5
2. [INSTALL\_DIR]/plugins/  
com.parasoft.xtest.sourcecontrol.eclipse.core\_<version> ディレクトリに  
ext というディレクトリを作成します。
  3. ext ディレクトリに統合パッケージをコピーします。



4. ツールを起動し、[Parasoft] > [設定] > [ソース管理] を選択して PTC Integrity オプションを利用できることを確認します。



## Parasoft DTP へのインストール

1. MKSjars.zip ファイルの内容を、DTP がインストールされているシステム中の一時的なディレクトリに展開します。
2. \$DTP\_HOME/plugins ディレクトリに scontrol というディレクトリを作成します。
3. 任意のエディターで \$DTP\_HOME/bin/reportserver.sh スクリプトを開きます。
4. CATALINA\_OPTS という行を探し、次の行を追加します。
 

```
-Dscontrol.ext.dir=$PST_HOME/plugins/scontrol \
```
5. スクリプト ファイルを保存し、エディターを終了します。

6. \$PST\_HOME/bin/concertoconsole.sh スクリプトを使って DTP を再起動します。

```
# Needs to be exported to be seen by catalina.sh
export JAVA_HOME
export JAVA_OPTS="-D$PCC_RECOGNITION"

export CATALINA_OPTS="-Djava.ext.dirs=$PST_HOME/lib/rs:$PST_HOME/lib/ext \
    $JAVA_MEM \
    -Dpcc.dir=$PST_HOME -Dinstall.dir=$GRS_HOME -Dtcn.dir=$TCM
_HOME -Dls.dir=$LS_HOME \
    -Dcom.parasoft.xtest.logging.config.file=$PST_HOME/bin/log
_config.xml \
    -Dscontrol.ext.dir=$PST_HOME/plugins/scontrol \
    -Dsource.scanner.log=source_scanner \
    -Djava.awt.headless=true"

export LICENSE_SERVER_HOME=$LS_HOME

if [ "$1" = "start" ] ; then

    checkPortAvailable

    echo "*** Starting catalina at" `date` >> $RS_OUT_LOG
    exec $CATALINA_HOME/bin/catalina.sh run >> $RS_OUT_LOG 2>> $RS_ERROR_LOG
"reportserver.sh" 148L, 3707C                               128,23          92%
```

## Parasoft Test での設定とセットアップ

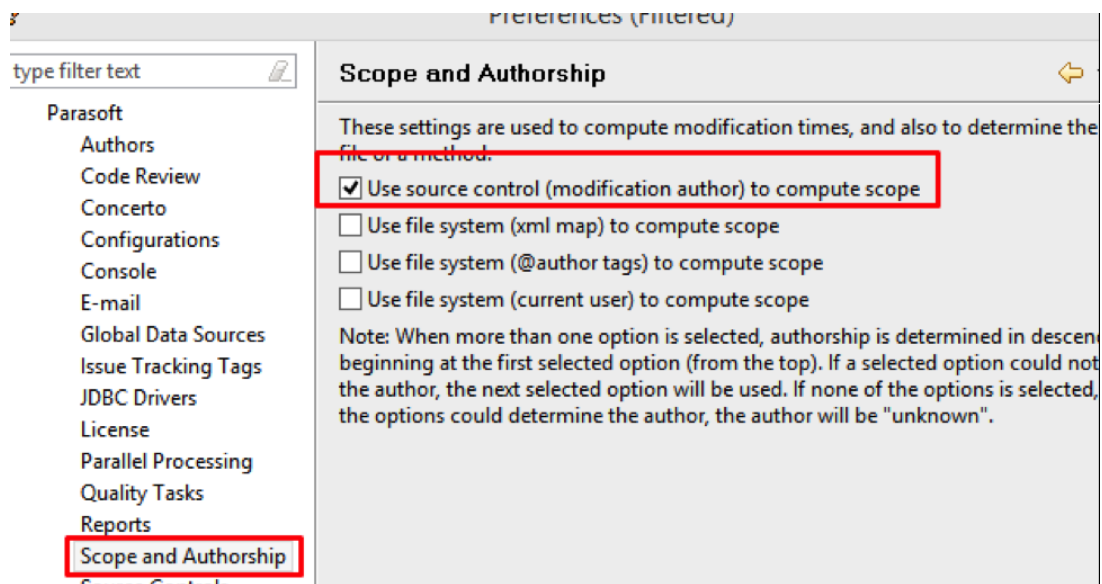
Linux/Solaris の場合、設定の残り部分に進む前に以下のステップを実行します。

- \$INSTALLDIR/bin ディレクトリを PATH の先頭に必ず追加します。
- \$INSTALLDIR/lib/linux (Solaris の場合、\$INSTALLDIR/lib/solaris) を指すように LD\_LIBRARY\_PATH を必ず設定します。

\$INSTALLDIR/lib/<linux/solaris> ディレクトリには 32-bit と 64bit の両方のライブラリがあります。JVM に基づいて MKS API が適切なライブラリをロードします。

1. Parasoft Test アプリケーションを起動します。
2. [Parasoft] > [設定] > [スコープと作成者] を選択します。
3. [ソース管理システム (更新作成者)] を使ってスコープを計算] チェックボックスをオンにします。

4. 他のすべてのチェックボックスをオフにして [適用] クリックします。



5. [Parasoft] > [設定] > [Parasoft] > [ソース管理] を選択し、PTC Integrity オプションが有効であること、および si コマンドが PATH に設定されていることを確認します。si が PATH がない場合、必ず絶対パスで si を PATH に追加してください。
6. [カスタム リポジトリ] セクションで [新規] をクリックし、[プロパティ] ウィンドウに次のオプションを入力します。

serverport: PTC Integrity ポート (デフォルトは 7001)

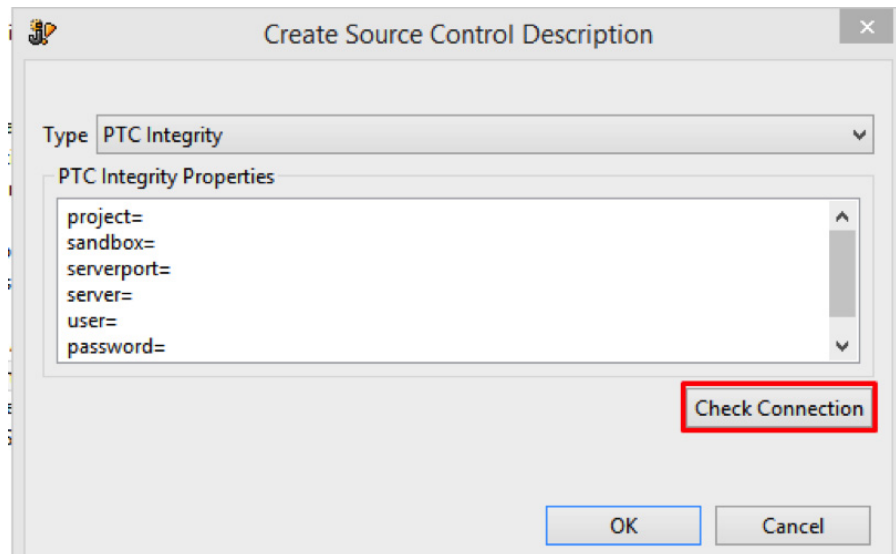
server: PTC Integrity サーバー名

user: PTC Integrity に対して認証されるユーザー名

password: PTC Integrity に対して認証されるパスワード

sandbox: ローカル リソースが同期化されるサンドボックスの場所 (パス中で project.pj で終わる)

project: サーバー上の PTC Integrity プロジェクトの場所 (例: /Project XZY/project.pj)



7. [ 接続のチェック ] をクリックし、オプション設定を確認します。エラーがある場合、PTC Integrity のエラーメッセージが返され、ダイアログに表示されます。
8. [OK] をクリックして、設定を保存して終了します。

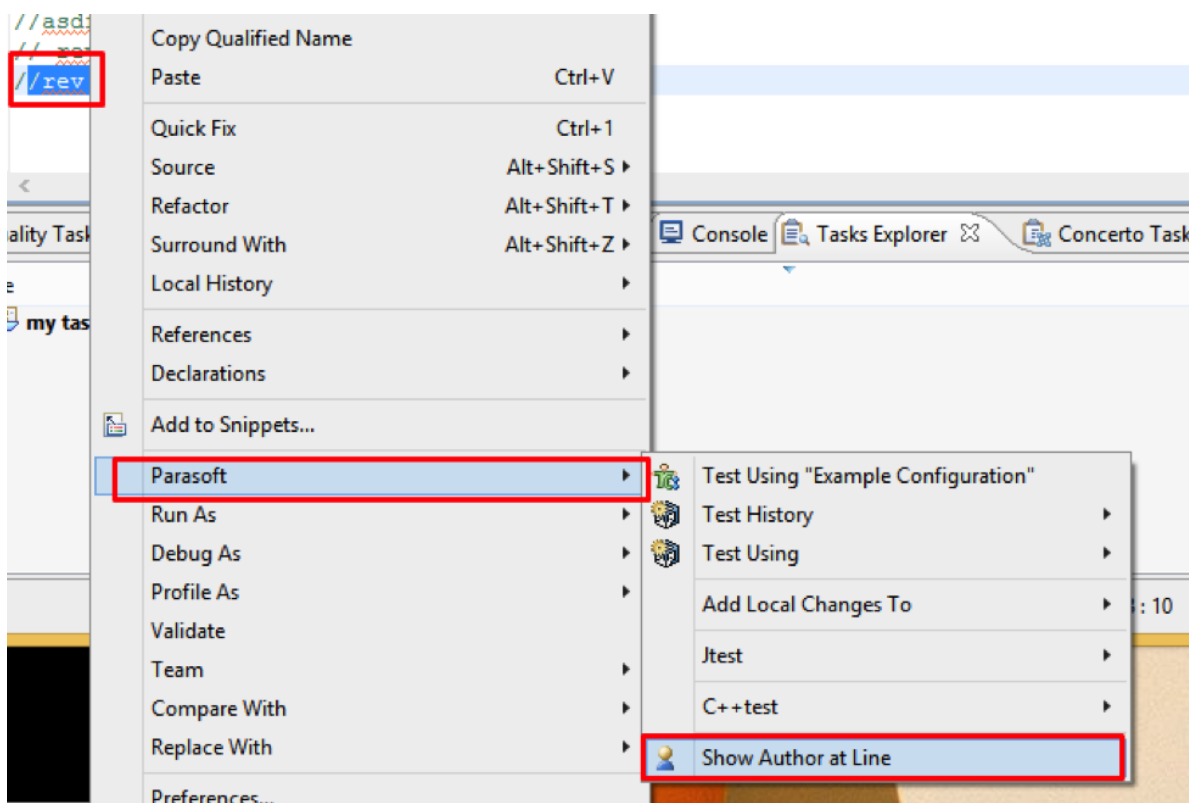
---

## Parasoft Test での設定の確認

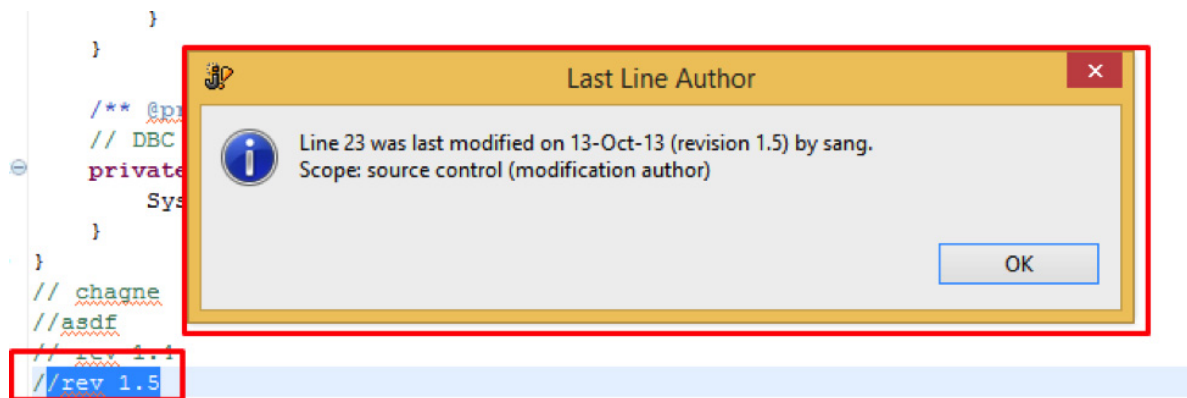
1. サンドボックスのプロジェクトを Parasoft Test ソリューションにインポートします。
2. ファイルを開き、行を右クリックします。



3. [Parasoft] メニューの [Show Author at Line] をクリックします。



4. ダイアログが開き、作成者の情報とリビジョン番号が表示されるはずですが。



## DTP と SourceScanner での PTC Integrity の設定

設定に進む前に、SourceScanner と DTP を実行するマシンに PTC Integrity (MKS) クライアントがインストールされていて正しく動作していることを確認してください。

Linux/Solaris の場合、設定の残り部分に進む前に以下のステップを実行します。

- \$INSTALLDIR/bin ディレクトリを PATH の先頭に必ず追加します。
- \$INSTALLDIR/lib/linux (Solaris の場合、\$INSTALLDIR/lib/solaris) を指すように LD\_LIBRARY\_PATH を必ず設定します。

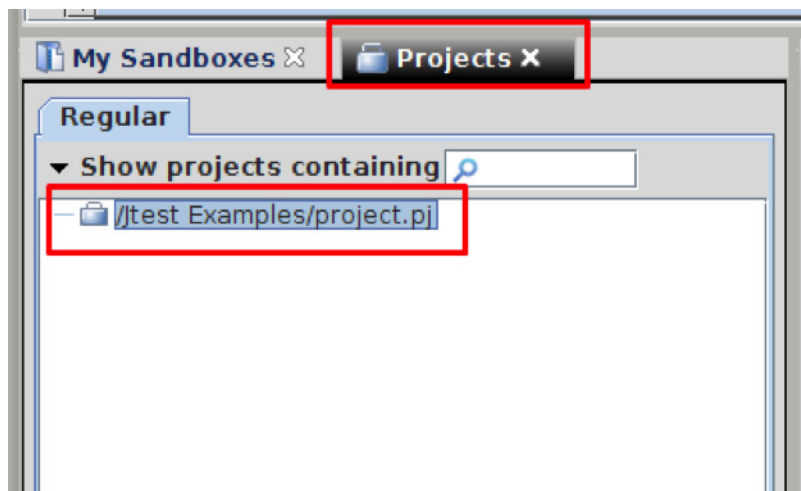
`$INSTALLDIR/lib/<linux/solaris>` ディレクトリには 32-bit と 64bit の両方のライブラリがあります。JVM に基づいて MKS API が適切なライブラリをロードします。

Windows の場合、SYSTEM 環境の PATH に必ず `%INSTALLDIR%/bin` を追加してください。

Bash の場合：

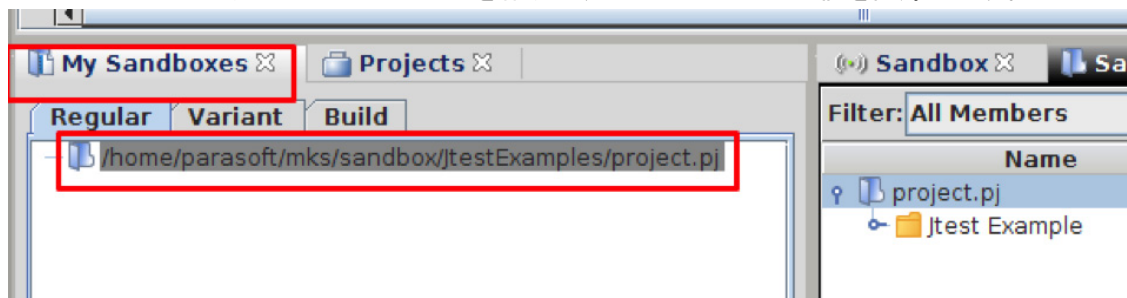
```
PATH=/home/Integrity/IntegrityClient10/bin:$PATH
LD_LIBRARY_PATH=/home/Integrity/IntegrityClient10/lib/
linux:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH PATH
```

1. `si gui` コマンドを開始して Integrity Client GUI を開きます。
2. [Projects] タブの Identify プロジェクトパスが次のように表示されます。



この値は、SourceScanner で Project Root として使用されます。

3. このシステム用のサンドボックスを作成し、サンドボックスの値を記録します。



この値は、SourceScanner でサンドボックスの場所として使用されます。

4. DTP のインストール ガイドに従って、DTP をインストールします。Linux で DTP を起動する前に、PATH と LD\_LIBRARY\_PATH が SHELL にロードされることを確認してください。
5. `SDLCExtensions-5.7.3.zip` ファイルを展開して SourceScanner をインストールします。

Windows の場合：

1. zip ファイルを `C:\Parasoft` に展開します。`C:\Parasoft\SDLCExtensions` ディレクトリが作成されます。

2. PROSERVE\_HOME 環境変数を C:\Parasoft\SDLCExtensions に設定します。

Linux/Solaris の場合 :

1. \$HOME/parasoft ディレクトリに zip ファイルを展開します (\$HOME/parasoft/SDLCExtensions ディレクトリが作成されます)。
2. bin ディレクトリに移動し、すべてのスクリプト実行可能ファイルに対して `chmod a+x *.sh` を実行します。
3. \$HOME/.proserve.rc を編集し、次の行を追加しました。

```
export JAVA_HOME=<JAVA_RUNTIME_INSTALL_DIR>
export PROSERVE_HOME=<where SDLCExtensions are installed>
($HOME/parasoft/SDLCExtensions)
```
4. DTP または SourceScanner を起動する前に、\$HOME/.bashrc で PATH および LD\_LIBRARY\_PATH が保存されてロードされることを確認します。
6. SSGUI.sh (Linux/Solaris) または SSGUI.cmd (Windows) を開始し、SourceScanner GUI を起動します。
7. SDLCExtensions のドキュメントに従って、SourceScanner をセットアップします。以下のセクションでは、PTC Integrity に関連する情報だけを説明します。

---

## PTC (MKS) Integrity プロジェクトを作成する

1. プロジェクトの下で、ソース管理の種類として MKS を選択します。
2. プロジェクト名を入力します。これはリポジトリ ID として DTP で使用されます。
3. ルートは、前述のステップ 2 で得られたサーバーのプロジェクト パスになります。
4. サンドボックスの場所は、前述のステップ 3 の値になります。
5. その他の設定と情報については、『SDLCExtensions ユーザーズ ガイド』を参照してください (注意: SDLCExtensions ドキュメントの MKS の参考情報は古い内容であり、6.0 リリースで更新される予定です)。
6. SourceScanner 実行してデータを DTP にポピュレートします。

---

## サードパーティのツールから DTP へのテスト結果の送信

DTP には、カスタムおよびサードパーティのテスト結果を DTP に送信するために使用できるオープン API が用意されています。テスト結果を DTP に送って、たとえば Report Center ダッシュボードで可視化するなど、開発プロセスでテスト結果を利用することができます。

この API はオープンソースのバンドルとして DTP に付属しています。詳細については次の URL を参照してください。

<http://sourceforge.net/projects/psfconcertoapi/?source=directory>

## ライセンスの変更

ライセンスを変更したい場合、テクマトリックスから新しいライセンス パスワードを入手して、新しいライセンスを設定できます。

1. License ページ下部の [Configure license] リンクをクリックします。  
[License Configuration] ページが表示されます。

Concerto のライセンス

---

ライセンスのステータス: [ライセンス OK](#)

---

Concerto のバージョン: Concerto 4.8  
マシン ID:

ライセンスの詳細:

有効期限:

パスワード:

機能:

- Concerto
- Concerto Report Center
- Concerto Project Center
- Concerto Code Review Server
- Concerto Policy Center
- Concerto Process Center
- Concerto Team Server
- Concerto License Server
- Concerto License Server borrowing

2. 取得したライセンスの有効期限とパスワードをそれぞれ [有効期限] および [パスワード] フィールドに入力します。
3. [ライセンスの設定] リンクをクリックしてライセンスを設定します。

# DNS サービスの設定

DNS Service は、特定のホストで Team Server、License Server、および Report Center のサービスが利用可能かどうかを表示するユーティリティです。DNS Service の設定は任意です。

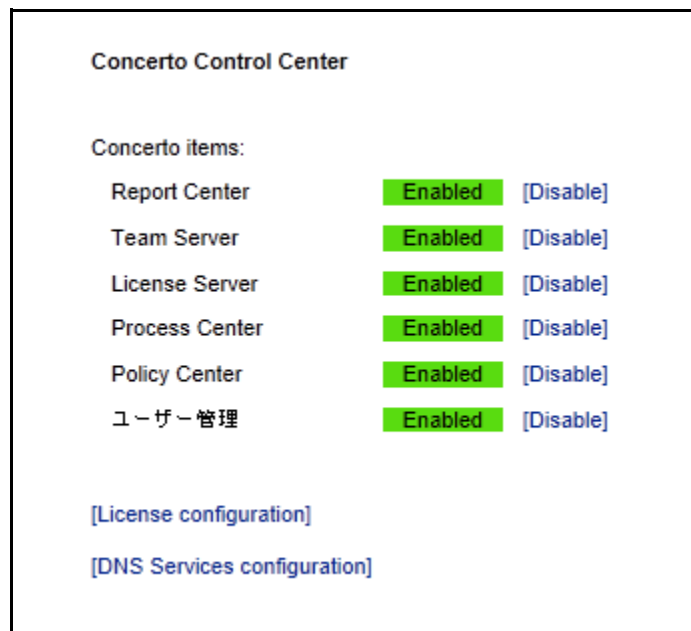
**注意：** Parasoft 製品でサーバーの自動検出機能を利用するには、DNS Service の設定が必要です。

DNS Service を設定するには、次の操作を行います。

1. DTP メイン ページの左上にある Parasoft ロゴをクリックします。



[Development Testing Platform Control Center] ページが表示されます。



2. [\[DNS Services Configuration\]](#) リンクをクリックします。  
[Service DNS Configuration Center] ページが表示されます。
3. デフォルトでは、Host IP にアクセスしているホスト名が表示されます。必要に応じて、ドロップダウン リストから設定を行うホストを選択します。

デフォルトのホスト IP が表示されています。DTP サービス起動マシンに複数のホスト IP が設定されている場合、必要に応じて [Host IP] ドロップ ダウン リストからホスト IP を選択します。

4. リストされているサービスのうち、自動検出するサービスに対して次の作業を行います。

- a. 名前を入力します ( 名前は他のサービスと識別するため変更可能です )。
- b. 優先順位を指定します。

**注意：** [Priority] フィールドは、[ 設定 ] ダイアログの [ チーム ] > [ 自動検出されたサーバー ] にサーバーを優先的に表示するためのものです。[ 自動検出されたサーバー ] がないクライアント ツールに対しては、この設定は無効です。

5. [Update] リンクをクリックして変更を適用します。

---

# MySQL サーバーのアップグレード

既存の Report Center データベースがあって、このデータベースを残したい場合に MySQL サーバーをアップグレードすることができます。アップグレードの前に、必ず Report Center データベース ファイルをバックアップしてください。Report Center データベースのバックアップの詳細については『Development Testing Platform クイック インストール ガイド』を参照してください。MySQL をアップグレードする方法については、MySQL ドキュメント ライブラリを参照してください。

<http://dev.mysql.com/doc/>



---

# Configuration Manager の内部詳細 (Linux/Solaris)

このセクションでは、Configuration Manager コマンドライン プログラムを使用するときに実行される、内部的な DTP スクリプト (DTP\_HOME/bin/dtpconsole.sh) について説明します。

このセクションの内容：

- Web サーバーの実行
- Data Collector の実行

---

## Web サーバーの実行

DTP Web サーバーを実行するために (1) を選択したとき、dtpconsole.sh は次のスクリプトを実行します。

- DTP\_HOME/bin/reportserver.sh

構成ファイル \$DTP\_HOME/.server\_params も次の行によって更新されます。

- service=ENABLED

エラー メッセージを含め、情報を持つ DTP 製品のすべてのログは、DTP\_HOME\logs\ ディレクトリに保存されます。メイン ログ ファイルは rs.log です。DTP Server のプロセス ID は DTP\_HOME/grs/log/rs.pid に格納されます。

他のログファイルの詳細については 641 ページの「検証ステップ 3: ログ ファイル中に未知の例外またはメッセージがない」を参照してください。

構成ファイルで service=ENABLED が定義されている限り、DTP Server プロセスの最中に問題が発生した場合、DTP のインストール時にインストールされた cron ジョブがプロセスをモニタリングし、DTP Server を再起動します。

---

## Data Collector の実行

Data Collector を実行するために (1) を選択したとき、dtpconsole.sh は次のスクリプトを実行します。

- DTP\_HOME/bin/datacollector.sh.

構成ファイル \$DTP\_HOME/.collector\_params も次の行によって更新されます。

- service=ENABLED

datacollector.sh スクリプトは Data Collector を実行します。エラー メッセージを含め、Data Collector のすべてのログは DTP\_HOME\logs\dc\* ファイルに保存されます。メインのログ ファイルは dc.log です。

Data Collector のプロセス ID は DTP\_HOME/grs/log/dc.pid に格納されます。構成ファイルで service=ENABLED が定義されている限り、データ収集プロセスの最中に問題が発生した場合、DTP

のインストール時にインストールされた cron ジョブがプロセスをモニタリングし、Data Collector を再起動します。

Data Collector が実行され、Parasoft 製品やサードパーティ製ツールから収集されたデータがデータベースに保存されます。

---

# Linux での初期データベースの作成とデータベースのアップグレード ( コマンドラインメニューからの実行 )

Linux で初期データベースを作成するには、次の操作を行います。

1. 次のコマンドを入力します。

```
dtppconsole.sh
```

スクリプトがパスに含まれていなければなりません。スクリプトは \$DTP\_HOME/bin ディレクトリにあります。

コマンドを実行すると、次のように Development Testing Platform Configuration Manager Main Menu が表示されます。

```
MAIN MENU
```

```
=====
```

```
Options:
```

```
    (1) Development Testing Platform Server (Report Center, Project  
        Center, Team Server, License Server)
```

```
    (2) Data Collector
```

```
    (3) Database (MySQL only)
```

```
    (4) Status
```

```
    (5) System Administration
```

```
    (q) Exit to system
```

```
Choose one:
```

2. プロンプトに「3」を入力して [Database] オプションを選択します。Main Menu の他のオプションの詳細については 868 ページの「Report Center の管理 (Linux)」を参照してください。

「3」を入力すると、Database Menu が表示されます。

```
DATABASE MENU
```

```
=====
```

```
Options:
```

```
    (1) Create initial database
```

```
    (2) Set mysql root password
```

```
    (q) Exit to previous menu
```

```
Choose one:
```

3. プロンプトに「1」を入力して [Create initial database] オプションを選択します。  
Database Menu の他のオプションの詳細については 869 ページの「MySQL の root ユーザー パスワードの変更」を参照してください。

**注意：**デフォルトでは、MySQL データベースは空の root パスワードでインストールされません。空の root パスワードではデータベース サーバーに接続できない場合、管理スクリプトは正しい MySQL root パスワードを入力するようプロンプトを表示します。

4. 新規インストールの場合、データベースをローカル マシンに作成するかどうかを訊ねられます。[Yes] を選択します。

**アップグレードの場合：**

アップグレードを実行するかどうかを訊ねられます。[Yes] を選択します。

プロンプトが表示され、root のパスワードを入力するよう求められます。

5. MySQL の構成で使用するパスワードを入力します。

スクリプトは root アカウントを使用して MySQL サーバーに接続し、初期データベースを作成するスクリプトを実行します。

**注意：**問題が発生した場合、以下のどちらかの可能性があります。

- 1) MySQL の構成に不足がある。
- 2) データベースの権限の問題がある。

6. ( オプション ) 念のためにレポートがすべて最新の状態かどうかを検証するよう要求できます。この操作は Report Center の Web インターフェイスで行います。変更の必要があるケースは少ないでしょう。

**アップグレードの場合：**

念のために次のコマンドを実行してもう一度データベースの整合性を検証すると良いでしょう。

```
mysqlcheck -u grs -pgrs GRS
```

7. ( オプション ) Report Center ディレクトリの容量を増やすため、またはその他の理由で MySQL 内にデータ ディレクトリを移動したい場合、ここで移動を行うと良いでしょう。

**注意：**MySQL へのアクセスで問題が発生した場合、次の SQL スクリプトを使用して初期データベースを作成できます。

```
DTP_HOME/grs/db/create_db_mysql.sql
```

この操作を行う時点で Report Center は正常にインストール済みでなければなりません。Report Center は DTP のインストール プロセスでインストールされます。

---

## crontab による自動起動の設定 (Linux)

通常、Linux で Development Testing Platform (DTP) をインストールする場合には crontab が変更されます。DTP Server を自動的に起動する `cronguard.sh` をシステム リブートのたびに呼び出すよう、crontab が構成されます。

サーバーの起動時における DTP の自動起動は crontab で実装されます。なぜなら、crontab ではルート アクセスが必要ないからです。UNIX の `/etc/init.d` サービスで実装したなら、ルート アクセスが必要になります。

DTP のインストール時に crontab の構成をスキップした場合、以下の操作を行って手動で crontab を設定してください。

1. DTP をインストールしたユーザーとしてログインします。
2. `crontab -e` コマンドを入力します。
3. エディターに以下を入力します。`$DTP_HOME` には、DTP がインストールされているディレクトリをフルパスで指定してください。

```
@reboot $DTP_HOME/bin/cronguard.sh > $DTP_HOME/logs/cronguard.log 2>&1
```

# トラブルシューティング

このセクションの内容：

- Report Center
- Cache Report Executor
- Team Server
- ソース表示機能
- デバッグ ログ モードへの切り替え

## ログ ファイルの参照

Report Center は柔軟性の高いツールです。デフォルト設定を再構成することは可能ですが、どうしても必要でない限り、再構成することは推奨しません。さらに、変更によってさらに調整が必要になることを十分に理解した上で、再構成を行うべきです。

すべてが正しく動作しているかを検証するために、ログに格納された情報を参照することができます。すべてのログは <DTP\_HOME>\logs\\*.＊ ディレクトリに格納されます。

次の表は、最も重要なログとその内容です。

ログ	内容
rs.log	サーバーからのログ情報。
recalculators.log	Report Center のバックグラウンド ジョブからの情報。各ジョブが今後実行されるスケジュール、ジョブの過去の実行からのログ、各ジョブの終了ステータスなどです。
dc.log	Data Collector からのログ情報。具体的には、時刻、Data Collector がレポートを受け取った IP 番号、レポートがデータベースに格納された開始時刻と終了時刻。

次の表は、その他のログとその内容です。

ログ	内容
analyze.log	JobAnalyzeBase バックグラウンド ジョブからのログ情報。このジョブは定期的に週に一度実行され、Report Center のデータベース テーブルを担当します。Report Center の各データベース テーブルの統計情報をリフレッシュするために、データベース テーブルの解析コマンドを実行します。
analyzeold.log	前回の JobAnalyzeBase の実行に対する analyze.log のコピーを格納します。
app_info.log	バージョンとビルド番号を格納した小さなテキスト ファイル。 [About] メニューからも参照できます。
bts_scanner.log	BTS Scanner バックグラウンド ジョブからのログ情報。BTS Scanner は、外部のバグ トラッキング システムと要件管理システムからをスキャンします。BTS Scanner は DTP_HOME/grs/config/bts に置かれ、バックグラウンドで定期的に行われます。詳細については 336 ページの「バグ トラッキング システムおよび要求管理システムの統合」を参照してください。
codereview.log	コード レビュー用の Team Server プラグインの内部ログ。Report Center はこのログを利用して Team Server からコード レビューのデータを取得します。

ログ	内容
conf_update.log	古いバージョンから新しいバージョンへの更新における、 /config/*.xml 構造の更新情報についての内部ログ。
cr_service.log	コード レビュー モジュール (クライアント マシンの Parasoft Test の コード レビュー クライアントのためのサーバー) からのログ。
db_update.log	古いバージョン から新しいバージョンへの更新における、Report Center データベース スキーマの更新情報についての内部ログ。
dc.stderr	(Data Collector の stderr) Data Collector のキャッチされなかった エラーを格納します。このログは空であるべきです。
dc.stdout	(Data Collector の stdout) 通常は空です。 <b>注意:</b> DC の実行からのすべての情報は dc.log に格納されます。
jms.log	JMS イベント ブローカーにパブリッシュされたすべてのイベントを含 みます。詳細については 853 ページの「DTP イベントのサブスクリプ ション」を参照してください。
ls.log	License Server (DTP アイテム) ログ ファイル。
ls_access.log	DTP License Server に Parasoft ツールが送るライセンス要求のリス ト。 このログからの情報は毎晩 Data Collector に送られ、Report Center の [ ツールの使用 ] レポートにこのデータが表示されます。
pst.log	DTP からの全般的なログ。
pstsec.log	User Administration モジュールからのログ。
recalculators.log.NN	(N は 1 から 10 までの数字) recalculators.log のアーカイブ版。 recalculators.log は、50MB に達すると recalculators.log.1 に移動 され、recalculators.log の内容は消去されます。10 個までアーカイ ブが作成されて保存されます。それを超えると、ログ情報は保存されま せん。
rs.log.NN	(N は 1 から 10 までの数字) rs.log のアーカイブ版。
tcm.log	Contains logs from Team Server モジュールからのログ。
tomcat.log	Apache Tomcat からのメイン ログ。



---

# Cache Report Executor

デフォルトでは、staticLinksConfig.xml で指定されたレポートは午前 5 時にキャッシュされ始めます。午前 8 時までにキャッシュされないレポートがある場合、次の点を確認してください。

- レポート キャッシュがオンになっているか？

245 ページの「データベース接続の変更」および 596 ページの「Report Center データ キャッシュの有効化 / 無効化」を参照してください。

- CacheReportExecutor ジョブの allowFromHour の設定はどうなっているか？

CronConfig.xml ファイルを確認してください。

- 今朝 CacheReportExecutor は起動したか？

DTP\_HOME\logs\recalculators.log を確認してください。このログ ファイルには Report Center のすべてのバックグラウンド ジョブが記録されます。

このファイルの最後に移動し、「CacheReportExecutor」を上に向かって検索します。

CacheReportExecutor は、staticLinksConfig.xml で指定されたレポートを実行するとき「Generating cache for」で始まるテキスト行をログに記録します。そのため、「Generating cache for」で検索することもできます。

- CacheReportExecutor は起動したかもしれないが、Report Center のレポートを参照するユーザーによって一時停止された。そして最後のユーザー アクションの後に待機時間が過ぎてまた動き出した可能性がある。

夜間にキャッシュする必要があるレポートが staticLinksConfig.xml で指定されていない場合、そのレポートを staticLinksConfig.xml に追加してください。詳細については 290 ページの「キャッシュするレポートの追加」を参照してください。

# 索引