



PI DataLink 2023 ユーザーガイド

© 2015-2023 by AVEVA Group plc or its subsidiaries. All rights reserved.

本書のいかなる部分も、AVEVA Group plc からの事前の書面による許可なく、機械的、複写機、記録などいかなる方法、いかなる形でも、転送、複製、検索システムへの登録を行うことはできません。ここに記載されている情報の使用に関しては、いかなる責任も負いません。

このマニュアルの準備には予防措置が取られていますが、AVEVA はエラーや省略に対して責任を負いません。このマニュアルに記載されている情報は予告なしに変更される場合があり、AVEVA 側の誓約を示すものではありません。このマニュアルに記載されているソフトウェアは、ライセンス契約に基づき提供されます。本ソフトウェアは、そのようなライセンス契約の条件に従ってのみ使用またはコピーできます。AVEVA、AVEVA ロゴおよびロゴタイプ、OSIsoft、OSIsoft ロゴおよびロゴタイプ、ArchestrA、Avantis、Citect、DYNSIM、eDNA、EYESIM、InBatch、InduSoft、InStep、IntelaTrac、InTouch、Managed PI、OASyS、OSIsoft Advanced Services、OSIsoft Cloud Services、OSIsoft Connected Services、OSIsoft EDS、PIPEPHASE、PI ACE、PI Advanced Computing Engine、PI AF SDK、PI API、PI Asset Framework、PI Audit Viewer、PI Builder、PI Cloud Connect、PI Connectors、PI Data Archive、PI DataLink、PI DataLink Server、PI Developers Club、PI Integrator for Business Analytics、PI Interfaces、PI JDBC Driver、PI Manual Logger、PI Notifications、PI ODBC Driver、PI OLEDB Enterprise、PI OLEDB Provider、PI OPC DA Server、PI OPC HDA Server、PI ProcessBook、PI SDK、PI Server、PI Square、PI System、PI System Access、PI Vision、PI Visualization Suite、PI Web API、PI WebParts、PI Web Services、PRISM、PRO/II、PROVISION、ROMeo、RLINK、RtReports、SIM4ME、SimCentral、SimSci、Skelta、SmartGlance、Spiral Software、WindowMaker、WindowViewer、Wonderware は AVEVA および/または子会社の商標です。その他のすべてのブランドは、それぞれの所有者の商標である可能性があります。

政府の権利

米国政府による使用、複製、開示は、AVEVA Group plc または子会社とのライセンス契約および DFARS 227.7202、DFARS 252.227-7013、FAR 12-212、FAR 52.227-19、またはこれらを継承するものに記載されている制限に準じるものとします。

発行日 : Monday, March 20, 2023

Publication ID(発行 ID): 1189207

お問い合わせ先

AVEVA Group plc
High Cross
Madingley Road
Cambridge
CB3 0HB.UK

<https://sw.aveva.com/>

セールスおよびカスタマートレーニングへの連絡方法については、<https://sw.aveva.com/contact> を参照してください。

テクニカルサポートへの連絡方法については、<https://sw.aveva.com/support> を参照してください。

AVEVA のナレッジベースとサポートセンターには、<https://softwaresupport.aveva.com> からアクセスできます。

目次

章 1 PI DataLink	8
章 2 はじめに.....	9
PI DataLink 概要.....	9
本リリースについて.....	9
PI DataLink のシステム要件.....	10
アップグレード時の互換性.....	10
サポートされるデータタイプ.....	10
PI DataLink のインストール.....	10
PI DataLink のインストール.....	11
サイレントインストール.....	11
サポート言語での PI DataLink の表示.....	12
PI Data Archive コレクティプの接続の優先.....	12
PI Data Archive の設定.....	12
Microsoft Excel アドイン設定.....	13
アドインの状態を表示する.....	13
無効なアドインを有効にする.....	13
アクティブでないアドインをアクティブにする.....	13
セキュリティの推奨事項.....	14
Excel ファイルセキュリティ.....	14
Excel アドインセキュリティ.....	15
PI DataLink 証明書の信頼済み証明書ストアへの追加.....	15
アカウント権限.....	15
組織のセキュリティ対策.....	15
ログの活用.....	16
本ガイドについて.....	17
章 3 概要.....	18
ユーザーインターフェイス.....	18
[PI DataLink] タブ.....	18
関数タスクペイン.....	18
コンテキストメニュー.....	19
PI DataLink 関数の概要.....	20
データアイテム.....	20
入力項目.....	21

時間入力.....	22
追加データ.....	23
表示書式.....	23
内挿値.....	24
検索.....	25
データアイテムの検索.....	25
ブラウズペインを使った検索の絞り込み.....	27
検索範囲バスを使った検索の絞り込み.....	28
フィルタリングによるアセットの検索.....	29
PI DataLink 設定.....	31
Excel の PI DataLink 設定を管理する.....	31
タイムゾーン設定の制限.....	34
コンピューター上のすべてのユーザーについて PI DataLink の設定を変更する.....	35
コンピューター上のすべてのユーザーから PI DataLink の設定を削除する.....	36
サーバー接続の管理.....	37
章 4 ワークシートの作成.....	38
ワークシートの作成プロセス.....	38
大規模データの取得.....	39
資産相対画面.....	39
資産相対画面の作成.....	40
共通の構造を持つデータアイテムのワークシートへの取り込み.....	40
挿入されたデータアイテムを参照する PI DataLink 関数の構成.....	41
ワークシートのイベント.....	42
エレメントに関連するイベントの検索.....	44
[イベントの検索]タスクペインへの属性列の追加.....	46
子イベントのあるイベントの検索.....	46
子イベントを含めることによるイベントの比較.....	48
子イベント属性を列として[イベントの比較]タスクペインに追加.....	50
親イベント属性を列として[イベントの比較]タスクペインに追加.....	53
親イベントを含めることによるイベントの比較.....	54
特定の親イベントのイベントを検索する.....	55
予約属性名.....	56
章 5 ワークシートの使用法とメンテナンス.....	58
関数配列.....	58
共通関数配列タスク.....	58
関数配列の入力の更新.....	59
配列の大きさ.....	60
配列のサイズ変更.....	60
動的関数配列.....	60
動的配列の挙動.....	61
動的配列に変換する.....	61
演算頻度.....	62
自動更新機能.....	62

自動更新の有効化.....	63
自動更新の無効化.....	63
トリガーされた再計算.....	63
揮発性 Excel 関数を含む PI DataLink 関数.....	63
手動による再計算.....	64
ワークシートの共有.....	64
[ブックの計算]オプションの[手動]設定.....	64
章 6 PI DataLink 関数.....	65
現在値関数.....	65
現在値関数の例.....	66
ヒストリ値関数.....	67
ヒストリ値関数の例.....	69
圧縮データ関数.....	69
圧縮データ関数の例.....	72
定間隔圧縮データ関数.....	73
定間隔ヒストリデータ関数の例.....	75
時間指定圧縮データ関数.....	76
時間指定ヒストリデータ関数の例.....	77
演算データ関数.....	78
演算データ関数の例.....	83
条件指定時間データ関数.....	84
条件指定時間データ関数の例.....	86
イベントの検索関数.....	87
[イベントの検索]タスクペインリファレンス.....	88
イベントの検索の例.....	92
イベントの比較関数.....	92
[Compare Events]タスクペインリファレンス.....	93
イベントの比較関数のパス注釈.....	98
イベントの比較の例.....	98
アセットフィルター検索関数.....	99
プロパティ関数.....	100
プロパティ関数の例.....	101
章 7 高度なトピック.....	103
PI 時間.....	103
PI 時間の省略形.....	103
PI 時間表記.....	104
タイムスタンプ指定.....	104
時間間隔の仕様.....	105
式.....	106
PI DataLink 関数での式の使用.....	107
式の構文.....	107
Excel のセル内の式.....	107

式でのセル参照.....	108
式のデータアイテムに関する制限.....	109
式の例.....	110
フィルター式.....	110
関数の手入力.....	110
関数の手動定義.....	111
引数の手動入力に関するガイドライン.....	111
表示コード.....	112
データ書き出し用の関数.....	114
PI Data Archive や PI AF にデータを書き込む.....	114
章 8 関数リファレンス.....	116
単一値関数.....	116
PICurrVal().....	116
PIArcVal().....	117
PIExpVal().....	118
PITagAtt().....	119
複数値関数.....	120
PINCompDat().....	120
PINCompFilDat().....	121
PICompDat().....	122
PICompFilDat().....	123
PISampDat().....	125
PISampFilDat().....	126
PIExpDat().....	128
PITimeDat().....	129
PITimeExpDat().....	130
計算関数.....	130
PIAdvCalcVal().....	131
PIAdvCalcFilVal().....	132
PIAdvCalcExpVal().....	134
PIAdvCalcExpFilVal().....	136
PIAdvCalcDat().....	138
PIAdvCalcFilDat().....	140
PIAdvCalcExpDat().....	142
PIAdvCalcExpFilDat().....	144
PITimeFilter().....	146
PITimeFilterVal().....	147
入力関数.....	148
PIPutVal() PIPutValX().....	148
廃止された関数.....	149
章 9 トラブルシューティング.....	150
ログファイル.....	150
PI DataLink タブが使用できない.....	150

配列とセルの制限.....	150
データ取得制限.....	151
関数でサポートされないデータアイテム.....	151
重複イベントの存在.....	151
セキュリティ.....	151
OIDC 認証との接続.....	152
PI Data Archive の接続に関する問題.....	154
認証プロトコルの有効化.....	155
明示的ログインプロンプトを許可するためのコンピューター設定.....	156
接続管理からログイン資格情報を入力する.....	156
タイムアウト例外エラーの解決.....	157
更新されないデータ.....	157

章 1

PI DataLink

PI DataLink 2023 には、PI Server 2023 以降で使用される場合に OIDC 認証のサポートが導入されています。リリースノートでこのリリースの修正、機能拡張、非推奨に関する詳細をお読みください。

注：PI DataLink 2023 は、Office 2016、2019、2021、Office 365（デスクトップ版）と互換性があります。

章 2

はじめに

PI DataLink によるこそ！このセクションでは、PI DataLink の概要と本リリースの新機能を説明し、さらにインストールと設定についても解説します。

PI DataLink 概要

PI DataLink は、PI Server からワークシートに直接データを表示する Microsoft Excel アドインです。Microsoft Excel の計算機能、図表機能、書式機能を組み合わせることで、PI DataLink は、PI System データの収集、監視、解析、報告のための強力なツールとなります。

PI DataLink 2023 は、OIDC 認証のサポートを PI Server に導入します。my.osisoft.com のリリースノートで、このリリースの修正、機能拡張、非推奨に関する追加詳細をご確認ください。詳細については、[Microsoft サポート](#)をご確認ください。

注: PI DataLink 2022 は Office 2013、2016、2019、2021、365 と互換性があります。

PI DataLink には、次の機能が備わっています:

- PI Data Archive Server から PI ポイント値を取得する
- PI Asset Framework (PI AF) Server から属性値を取得する
- システムメタデータを取得して、PI System データを構造的に表示する:
 - PI ポイント名と属性
 - PI AF 属性とエレメント
- PI DataLink 関数を使用して上記のアイテムを参照し、データを計算してフィルターする
- ワークシート再計算時に取得値を更新する

PI DataLink には、ユーザーが関数を作成してデータを取得するためのユーザーインターフェイスが含まれます。PI DataLink は関数をワークシートのセルに埋め込み、PI System からリアルタイムデータを使用して随時更新します。

さらに Excel の豊富な計算機能や書式機能を使用し、目的や閲覧者に合わせて PI システムデータを適切な形式で表示できます。

本リリースについて

このセクションでは、システム要件、以前のリリースからのアップグレード時の互換性、サポートされているデータタイプについて説明します。

PI DataLink のシステム要件

システム要件については、[OSIsoft カスタマーポータルの PI DataLink のリリースノート](#)をご確認ください。

アップグレード時の互換性

PI DataLink は下位互換性があります。本バージョンの PI DataLink では、旧バージョンで作成したワークブックを、変換することなく読み取れます。しかし、この逆の場合が可能とは限りません。ワークシートを新バージョンで変更すると、旧バージョンの PI DataLink では作業できなくなる可能性があります。

このバージョンの PI DataLink は、データアクセスに PI AF SDK のみを使用します。PI DataLink の初期バージョンもデータアクセスに使用している PI SDK は、使用されなくなります。

サポートされるデータタイプ

PI DataLink は、以下の PI ポイントデータタイプをサポートします:

- Digital (定義されたステータス)
- Int (16、 32)
- Float (16、 32、 64)
- String (テキスト)
- Timestamp

PI DataLink は BLOB タイプをサポートしていません。

PI DataLink は、次の PI 属性値タイプをサポートします:

- Byte
- Int (16、 32、 64)
- Single
- Double
- String*
- DateTime*
- Boolean*
- Enumeration*

*演算データ関数ではサポートされていません。

PI DataLink は、PI AF 属性値タイプ、GUID、属性、エレメント、ファイル、配列をサポートしていません。

PI DataLink のインストール

PI DataLink を使用する前に、配布 CD か OSIsoft からダウンロードしたセットアップキットで PI DataLink をインストールする必要があります。必要に応じて、サイレントインストールを使用できます。Office の言語パックをインストールして Office の表示言語を変更し、英語以外の言語に対するサポートを有効

化できます。ご使用中の PI Data Archive 構成を考慮して、PI ポイントに正常にアクセスできることを確認してください。

PI DataLink のインストール

セットアッププログラムを起動し、PI DataLink をインストールします。セットアッププログラムは 32 ビット版と 64 ビット版の PI DataLink の両方を自動的にインストールします。1 つのバージョンのみをインストールする場合は、**setup.ini** ファイルを修正してください。旧バージョンの PI DataLink がインストールされている場合は、インストーラーが自動でアップグレードを実行し、旧バージョンからの設定を引き継ぎます。

1. セットアップキットを用意します。次の操作を実行できます。

- <https://my.osisoft.com/> からキットをダウンロードします。
[ダウンロード] ページをフィルターして、インストールキットを表示できます。
- 配布 CD を挿入します。

2. コンピューターでファイルを解凍します。

3. 必要に応じて、次のことを行うために **setup.ini** ファイルを変更します。

- PI DataLink(32 ビットまたは 64 ビット)の 1 つのバージョンをインストールする。
setup.ini ファイルを変更するには、ファイル内の例に従ってください。

4. **setup.exe** プログラムを起動します。

セットアッププログラムは、PI DataLink をルートディレクトリ **PIPC**(通常、以下の場所)にインストールします:

C:\Program Files\PIPC\Excel

プログラムは **./PIPC/Help** ディレクトリの対応する言語ディレクトリにオンラインヘルプファイルをインストールします。 <https://my.osisoft.com> から PDF 版とリリースノートをダウンロードできます。

- デフォルトの PI DataLink の設定を変更する場合は、[コンピューター上のすべてのユーザーについて PI DataLink の設定を変更する](#) を参照してください。
- 英語以外の言語で PI DataLink を表示するには、[サポート言語での PI DataLink の表示](#) を参照します。

サイレントインストール

本ソフトウェアは、Windows サイレントインストール機能を使用してインストールできます。サイレントインストールは無人インストールと呼ばれることもあり、セットアップ中にユーザーによる入力を必要としません。ソフトウェア自動配布ツールとサイレントインストールを組み合わせれば、システム管理者はソフトウェアを多くの社内 PC に自動配布できます。

サイレントインストールを開始するには、次のように入力してください。

Setup.exe -f silent.ini

silent.ini ファイルはセットアップキットに含まれています。サイト毎に異なるファイル設定をすることもできます。設定項目に関する詳細情報と説明については、**silent.ini** ファイルを参照してください。

サポート言語での PI DataLink の表示

PI DataLink は複数の言語をサポートしています。PI DataLink がサポートしていない言語の場合、PI DataLink ユーザーインターフェイスは英語で表示されます。

Microsoft Excel で目的の言語を設定します。

1. Microsoft Office の言語パックをインストールします。[Microsoft サポート](#)をご確認ください。
2. Microsoft Excel の Office の表示言語を変更します。[Microsoft サポート](#)をご確認ください。
3. Excel を再起動します。言語がサポートされている場合は、選択した言語で PI DataLink が表示されます。サポートされていない場合は、引き続き英語で表示されます。

PI Data Archive コレクティブの接続の優先

PI DataLink が PI Data Archive コレクティブに接続する場合、コレクティブへの初回接続時に PI DataLink は接続設定を行います。設定は、接続を起動するコンポーネントによって異なります:

- PI DataLink 関数は Any を設定します。
- 検索は Any を設定します。
- Connection Manager ([設定] ウィンドウからアクセス) では、設定が Prefer Primary、または PI System Explorer の設定 (別途設定されている場合) になります。PI System Explorer の環境設定については、PI Server のトピック 「PI System Explorer の接続設定を管理する」 を参照してください。

PI Data Archive の設定

PI DataLink は、PI Data Archive 3.4.380 以降からデータを取得できます。ただし、PI DataLink は適切な PI Data Archive Server に接続し、ポイントに正常にアクセスできなければなりません。これには、以下のものに対する設定変更が必要になることがあります。

- ファイアウォールデータベース

各 PI Data Archive コンピューター上のファイアウォールデータベースは、PI DataLink を実行するクライアントコンピューターからのアクセスを可能にするような構成にする必要があります。

- 認証と承認

PI DataLink のユーザーは、PI Data Archive での ID を認証することが可能であり、PI Data Archive へのアクセス権限を持っている必要があります。ユーザーに読み取り専用権限のような必要最低限のアクセス権を割り当てます。認証については、PI Mapping (PI Data Archive バージョン 3.4.380 以降で使用可能) の使用をお勧めします。PI Trust や PI パスワード認証を使用することもできます。ただし、PI パスワード認証は安全性が低いため、PI パスワード認証の使用はお勧めしません。

- ポイント定義

許可されているユーザーに読み取り権限(必要に応じて書き込み権限)を与えるようにポイントを定義する必要があります。

詳しくは、PI Data Archive のマニュアルや、PI Server のトピック 「PI PI Data Archive のセキュリティ」 などを参照してください。

Microsoft Excel アドイン設定

PI DataLink は Microsoft Excel へのアプリケーションアドインです。

注: このアドインセクションの手順は、Microsoft Excel アプリケーションのローカルコピーの構成を参照しています。

名前	場所	タイプ
PI DataLink	...\\PIPC\\Excel\\OSisoft.PIDataLink.UI.vsto	COM アドイン

セットアッププログラムが PI DataLink アドインをインストールしてアクティブにします。

まれに、Microsoft Excel によりアドインが無効化される場合があります。アドインは、有効にしてからアクティブにする必要があります。

注: アプリケーションアドインをアクティブにするには、コンピューターの管理者権限が必要です。ユーザーが管理者ではない場合、Windows エクスプローラーで **Excel.exe** を右クリックし、次に**管理者として実行**をクリックし、管理者として Microsoft Excel を実行します。

アドインの状態を表示する

アドインの状態（アクティブ、アクティブでない、無効）を確認します。

1. [ファイル]タブをクリックし、[オプション]をクリックします。
2. [Excel のオプション]ウィンドウで[アドイン]をクリックします。
3. アドインの一覧で、アドインの現在の状態を確認します。

各アドインは、以下のボックスに一覧表示されます。

- アクティブなアプリケーションアドイン
- アクティブでないアプリケーションアドイン
- 無効なアプリケーションアドイン

無効なアドインを有効にする

アドインが無効な場合、アドインを有効にしてから、アクティブにする必要があります。

1. [ファイル]タブをクリックし、[オプション]をクリックします。
2. [Excel のオプション]ウィンドウで[アドイン]をクリックします。
3. [管理]リストから、[使用できないアイテム]を選択肢、[設定]をクリックします。
4. 目的のアドインの隣にあるチェックボックスを選択します。
5. [有効にする]をクリックします。

アクティブでないアドインをアクティブにする

アクティブでないアドインをアクティブにし、Microsoft Excel で使用できるようにします。

注: アプリケーションアドインをアクティブにするには、コンピューターの管理者権限が必要になります。ユーザーが管理者ではない場合、Windows エクスプローラーで **Excel.exe** を右クリックします。次に**[管理者として実行]**をクリックし、管理者として Microsoft Excel を実行します。

1. [ファイル]タブをクリックし、[オプション]をクリックします。
2. [Excel のオプション]ウィンドウで[アドイン]をクリックします。
3. [Manage (管理)]リストから、[Com Add-ins (COM アドイン)]を選択し、[Go (設定)]をクリックします。
4. 目的のアドインの隣にあるチェックボックスを選択します。
5. [OK]をクリックします。

セキュリティの推奨事項

PI System は保護する必要のある機密データを格納している場合があります。弊社は不正アクセスのリスクを最小限に抑えるよう製品を設計しています。このセクションでは、PI DataLink で作業する際、データセキュリティを最大限に高めるための推奨プラクティスについて説明します。

Excel ファイルセキュリティ

Excel ワークブックに、クライアントコンピューター上に存在する場合に保護される必要のある機密データが含まれることがあります。PI System が提供する基本的なセキュリティは、ワークシートに情報を取得した後では適用されません。このため、各ワークブックファイルを保護する必要があります。次の操作を実行できます。

- Excel ワークブックをパスワードで保護します。パスワードにより、許可されたユーザーのみがワークブックのデータを表示または修正することを許可されます。詳しくは、Microsoft Office の記事「[ブックを保護する](#)」をご覧ください。
- ワークブックのパスワードを設定するために、最も安全なプラクティスを行います:
 - ワークブックを開くパスワード、およびワークブックを修正するパスワードを設定します。
 - パスワードには大文字、小文字、数字、および記号の組み合わせを使用します。
 - [ドキュメントプロパティを暗号化する]チェックボックスを選択して、許可されていないユーザーがワークブックの概要とカスタムファイルのプロパティを表示できないようにします。
- Excel ワークブックで Information Rights Management (IRM)を有効にします。IRM により、ユーザーまたは管理者はワークブックのアクセス許可を指定することで、許可されていないユーザーが PI System の機密データを印刷、転送、またはコピーを行えないようになります。IRM によりファイルの権限を制限した後、アクセス許可がファイル自体に格納されるため、情報がどこにあるかに関係なくアクセス制限および使用制限は適用されます。詳細については、Microsoft Office の記事「[Office 2010 の Information Rights Management](#)」を参照してください。必要に応じて、Excel ファイルのデータの有効期限を設定することができます。詳細については、前述の Microsoft Office の記事「ファイルに有効期限を設定する」の手順を参照してください。
- Excel ワークブックをリモートドライブに格納する場合は、PI DataLink をホストするコンピューターと Excel ワークブックを含むファイルストア間のリンクにインターネットプロトコルセキュリティ (IPSec) 暗号化を使用します。詳しくは、Microsoft TechNet の記事「[What Is IPSec \(IPSec とは\)](#)」を参照してください。

- Excel ワークブックに安全なファイル許可を適用し、許可されていないユーザーがファイルの内容にアクセスできないようにします。詳細については、Microsoft TechNet の記事「[File and Folder Permissions](#)」を参照してください。
- Excel ワークブックにデジタル署名を適用します。詳細については、Microsoft Support の記事「[Excel のブックでのデジタル署名とコード署名について](#)」を参照してください。

Excel アドインセキュリティ

Excel セキュリティセンターを使用して、アドインの動作を管理できます。Microsoft Excel のワークシートで許可されていないソフトウェアが実行されないようにするために、アドインに対して、信頼できる発行元の署名を必須とすることが推奨されます。具体的には、[セキュリティセンター]の[Add-ins]ページを使用して、次の設定を行います:

- アプリケーションアドインに対し、信頼できる発行元の署名を必須にする設定します。
- 署名されていないアドインに関する通知を無効にする

PI DataLink を標準の **C:\Program Files** ディレクトリ外にインストールし、アドインに対して、信頼できる発行元の署名を必須とするように設定している場合、PI DataLink 証明書を手動で信頼済み証明書ストアに追加する必要があります。「[PI DataLink 証明書の信頼済み証明書ストアへの追加](#)」を参照してください。

詳細については、Microsoft Office の記事「[Office プログラムでアドインを表示、管理、インストールする](#)」を参照してください。

PI DataLink 証明書の信頼済み証明書ストアへの追加

アドインに対して信頼できる発行元の署名を必須とするように Microsoft Excel を設定し、PI DataLink を標準の **C:\Program Files** ディレクトリ外にインストールする場合、PI DataLink 証明書を手動で信頼済み証明書ストアに追加する必要があります。

1. 管理者としてコマンドウィンドウを開きます。
2. コマンドプロンプトで、PI DataLink 証明書(**pidlcert.cer**)を含むディレクトリに移動します。
証明書は、インストールフォルダの **Excel** サブフォルダ(**PIHOME** 環境変数で定義)内にあります。
3. 以下のコマンドを入力します。

```
C:\Windows\System32\certutil.exe -addstore TrustedPublisher pidlcert.cer
```

アカウント権限

悪意あるユーザーからの被害を避けるために、ユーザーに読み取り専用権限のような必要最低限のアクセス権を割り当てます。

組織のセキュリティ対策

攻撃者がシステムにアクセスするのを防ぐために、セキュリティ対策を採用する必要があります。

- 物理的にコンピューターを保護します。PI DataLink を実行しているコンピューターにアクセスする権限を入手した攻撃者は、そのコンピューターの Excel ファイルに取得および格納されたすべての PI System データにアクセスできます。

- 権限のある従業員やゲストのみに、コンピューターへのアクセスを制限します。紛失した、または盗難にあったコンピューターから組織のネットワークへのアクセスを防ぐ必要があります。
- 最新のセキュリティアップデートをすべてのコンピューターに適用します。セキュリティ通知サービスを配信登録し、OS およびコンポーネント用の最新のセキュリティアップデートの通知を受信するようにします。詳しくは、Microsoft TechNet の記事「[マイクロソフトテクニカルセキュリティ通知のご案内](#)」を参照してください。
- 権限のない管理者から保護します。権限のない管理者は、多くの攻撃を仕掛けることができます。たとえば、以下のような攻撃を行います：
 - 悪質なソフトウェアをインストールし実行します。
 - リモートアクセスを設定して、コンピューターをリモートコントロールできるようにします。
- すべての管理アクティビティを監査し、定期的に監査ログを確認します。すべての管理者に対し雇用する前に素行調査を行い、雇用条件として定期的に再調査を行う必要があります。
- 複数のセキュリティ層を提供します。ファイアウォールなどの境界セキュリティのみに依存すると、ファイアウォールが侵害された場合、リスクが高まります。安全性の高いクライアントと安全性の低いクライアントを分離するようネットワークに構成することで、別のセキュリティ層を提供できます。クライアントコンピューターの個人用ファイアウォールを使用すれば、付加的な層を追加できます。侵入検知ソフトウェア、および疑わしいアクティビティを除外するホストベースの侵入検知ソフトウェアにより、セキュリティはさらに強化されます。また、アンチウイルスソフトウェアを実行することは不可欠です。そして、コンピューターセキュリティについてユーザーを教育することは、ネットワークセキュリティ戦略において非常に重要です。
- すべてのシステムに対して安全なベースラインを作成し、維持します。コンピューターを構成し管理する方法の詳細な説明を各ベースラインに指定します。この説明には、保護されたコンピューターに関するすべての構成設定を含める必要があります。安全なベースラインを作成するには、可能な限り最も安全な OS を使用します。最新の OS であれば、セキュリティを考慮して設計されている可能性が高く、OS を保護する機能が搭載されています。利用可能になったセキュリティアップデートを適用して、OS およびアプリケーションを最新の状態に保つようにしてください。
- 強力なパスワードまたはパスフレーズを使用します。空白パスワードは決して使用しないでください。パスワードのコンセプトの詳細については、Microsoft TechNet の記事「[Account Policies](#)」を参照してください。
- PI DataLink Excel ファイルへのアクセスを制御します。アクセスコントロールプロシージャを実装することで、すべての Excel ファイルが安全に保存され、攻撃者がファイルの内容を変更したり、情報を検索したりできないようにします。

ログの活用

組織はログを適切に活用する必要があります。ログファイルは以下の用途で使用できます：

- 追跡。トランザクションに関する統計データを提供するためにログファイルを使用できます。
- 検証。トランザクションを再作成するために、すべてのトランザクションの記録を提供するようログファイルを使用できます。
- トランザクションの有効性の証明。

PI DataLink は、PI AF SDK を介して、すべてのデータリクエストを送信します。PI DataLink は、個別のログファイルは維持しませんが、PI AF SDK ログファイルおよび PI Data Archive ログファイルを使用して、

PI DataLink トランザクションの情報を取得できます。詳細については、PI AF SDK および PI Data Archive のドキュメントを参照してください。

本ガイドについて

PI DataLink User Guide には、製品機能の詳細説明と PI DataLink 関数の全リファレンスが記載されています。このドキュメントの PDF 版とオンラインヘルプ版の内容は同一です。

このガイドは、PI System に関する基本的な理解がユーザーにあることを前提としています。PI Data Archive、PI AF について詳しくは、各製品のマニュアルを参照してください。PI Data Archive や PI AF に関する知識は、PI DataLink で使用されている用語を理解する際に役立ちます。

またトレーニング資料も含まれていますのでご活用ください。こうした資料には、OSIsoft が PI DataLink のトレーニングコースで使用している演習も含まれています。

[OSIsoft Customer Portal](#) から他の OSIsoft 製品のトレーニング資料およびマニュアルをダウンロードできます。 <https://docs.osisoft.com/> でマニュアルを表示できます。

用語の変更

OSIsoft では、PI System が従来のシングルサーバー構成から拡張されたことに伴い、用語の改訂を実施しました。改定後の用語では、PI Data Archive は時系列データを格納するコンポーネント（従来の PI Server）を指し、PI Server は PI Data Archive と PI Asset Framework の両方を指します。このマニュアルでは改定後の用語が使用されています。

章 3

概要

PI DataLink を使用する際には、PI DataLink 関数を定義して、PI System データを Excel ワークシートの関数配列として抽出します。関数配列は、1 つの PI DataLink 関数の出力内容が含まれるセルのかたまりです。以下のセクションでは、基本的な PI DataLink の概要と機能を紹介します。以下のセクションを理解した後で、PI DataLink でワークシートの作成と PI System データの取得を行ってください。

以下のセクションの内容:

- ユーザーインターフェイス機能
- PI DataLink の操作に必要なコア機能
- 目的、ニーズ、リソースに応じて PI DataLink のワークシートを作成するための様々なアプローチ
- ビデオ

PI DataLink については、YouTube の OSIsoft 学習チャンネルで PI DataLink のプレイリストもご覧ください。

ユーザーインターフェイス

PI DataLink は、独自のメニュー命令、ウィンドウ、タスクペインを Microsoft Excel に追加します。以下のセクションでは、PI DataLink インターフェイスの基本要素を説明します。

[PI DataLink] タブ

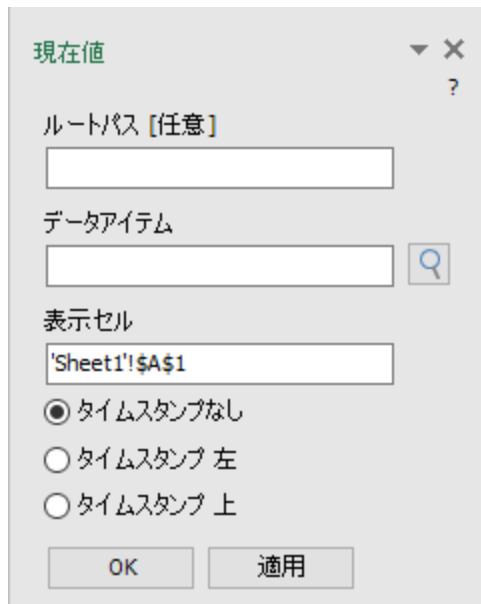
PI DataLink は、[PI DataLink] タブを Microsoft Excel リボンに追加します。



- [PI DataLink] タブをクリックすると、PI DataLink コマンドにアクセスできます。
- コマンドの 1 つをクリックすると、対応するタスクペインやウィンドウが表示されます。
- コマンドの上にカーソルをかざすと、説明がツールチップに表示されます。

関数タスクペイン

関数タスクペインで PI DataLink 関数を定義します。



タスクペインは、移動やドッキングが可能なコントロールのパネルです。タスクペインを開いている状態でも、ワークシートでの作業を続けることができます。

関数タスクペインを開くには、次の操作を行います:

- 出力セルをクリックし、[PI DataLink]タブ上の関数をクリックして関数を追加します。
- 既存の関数配列内のセルをクリックし、対応するタスクペインを表示して関数の引数を編集します。

注: タスクペインの自動表示を無効化することも可能です。「[Excel の PI DataLink 設定を管理する](#)」を参照してください。

- 手動で作業ウィンドウを表示するには、関数配列セルを右クリックし、関数名をクリックします。

開いた関数タスクペインで次の操作を行います:

- 入力値を入力または変更し、次の操作を行います:
 - [OK]をクリックすると、入力値の保存および生成された関数配列の表示が行われ、タスクペインが閉じます。
 - [Apply]をクリックすると、入力値の保存および生成された関数配列の表示は行われますが、タスクペインは閉じません。
- 矢印ボタンをクリックして、[Move]、[Size]、または[Close]コマンドを選択します。
- タイトルバーをドラッグすると、タスクペインを切り離せます。
- タイトルバーをダブルクリックすると、タスクペインをドッキングできます。
- カーソルをペインの端に置いてドラッグすると、サイズを変更できます。

コンテキストメニュー

PI DataLink は、Microsoft Excel 標準のコンテキストメニューに、いくつかのコマンドを追加します。これらのコマンドを使用して、ワークシート内の関数配列を管理できます。

PI DataLink コマンドが追加されたコンテキストメニューを開くには、PI DataLink 関数配列内の任意の場所を右クリックします。

PI DataLink は、コンテキストメニューに、次のコマンドを追加します:

- **Select DataLink Function**

関数配列全体を選択します。

新しいワークシートに配列をコピー、切り取り、移動する前に配列を選択してください。

- **Recalculate (Resize) Function**

関数配列全体を更新します。PI DataLink は PI Data Archive Server や PI AF から新しい値を取得し、取得したデータに合わせて配列のサイズを自動的に変更します。

- **Function Name**

対応する関数タスクペインを開き、関数配列の入力値を表示します。

- **動的配列に変換**

関数を動的配列に変換します

次の

[関数タスクペイン](#)

PI DataLink 関数の概要

PI DataLink 関数は、PI System データを Excel に抽出します。関数タスクペインを使用することで、関数を定義する入力値を簡単に入力できます。PI DataLink に慣れることで、Excel の数式バーに直接関数を入力することもできます。

以下のセクションの内容:

- PI DataLink 関数タスクペインの共通機能
- リアルタイムデータの取得/表示方法
- ワークシートにおける PI DataLink 関数の使用方法

次の

[関数タスクペイン](#)

[関数の手入力](#)

データアイテム

PI DataLink 関数は、指定した PI System データアイテム(PI ポイントまたは PI AF 属性)のデータを返します。関数により返される値は、データベースに記録されている値や、指定した基準や式を基に計算した値となります。

関数の定義時には、[データアイテム]、[式]、[フィルター式]フィールドと一緒に、データアイテムへのパスを[ルートパス]フィールド(任意)に指定します。以下のように指定できます:

- [データアイテム]、[式]、[フィルター式]項目では、完全修飾パスを使用します。完全修飾パスは 2 本のバックスラッシュ(\\)で始まり、PI ポイントまたは PI AF 属性として評価されます。

- 部分パスを[データアイテム]、[式]、[フィルター式]フィールドに、共通のベースパスを[ルートパス]フィールドに指定します。PI ポイント名や PI AF 属性名が部分パスの例になります。PI DataLink は共通のベースパスと部分パスを結合して、PI ポイントや PI AF 属性を表すフルパスを指定します。

PI DataLink がデータアイテムを参照するには、フルパスが必要になります。PI ポイントを参照するには、検索対象の PI Data Archive サーバーとポイント名を PI DataLink に指定する必要があります。PI AF 属性を参照するには、PI AF Server、データベース、エレメントの他にサブエレメントや親属性を PI DataLink に指定する必要があります。パイプ文字(|)を使用して、エレメントや親属性から属性を分離します。

[データアイテム]、[式]、および[フィルター式]フィールドが完全修飾パスや既定の PI Data Archive サーバーの PI ポイントを含む場合、[ルートパス]をブランクにできます。

[ルートパス]フィールドを指定した場合、ルートパスは、指定した複数のデータアイテムへの共通パスを示します。

データアイテムのタイプ	ルートパスフィールドの内容
PI ポイント	<p>ポイントを格納する PI Data Archive Server。または、既定の PI Data Archive Server を示すブランク。有効な項目は以下のものを含みます。</p> <ul style="list-style-type: none">• MyPIDataArchiveServer• \\MyPIDataArchiveServer
PI AF 属性	<p>PI AF Server、データベース、および[データアイテム]で指定していないエレメント、サブエレメント、親属性。有効な項目は以下のものを含みます。</p> <ul style="list-style-type: none">• \\MyPIAFServer\MyDatabase• \\MyPIAFServer\MyDatabase\MyElement• \\MyPIAFServer\MyDatabase\MyElement\MySubElement• \\MyPIAFServer\MyDatabase\MyElement\MyParentAttribute• \\MyPIAFServer\MyDatabase\MyElement\MyParentAttribute1\MyParentAttribute2 <p><u>注: PI AF Server とデータベースは、同じフィールドになければなりません。</u> [ルートパス]フィールドに PI AF Server に指定し、[データアイテム]フィールドにデータベースを指定することはできません。</p>

次の

[式](#)

入力項目

関数タスクペイン上のラベル付き入力値は、関数の適切な値を受け取ります。多くの場合、入力値には、値または値を含むワークシートセルへの参照を入力します。タスクペインを開いた際に、デフォルト値が入力されているものもあります。(任意指定) とマークされている入力値は必須ではありません。

編集フィールド を持った入力値では、次の操作を行えます：

- 文字列(PI AF 属性名や表示セルの場所など)を編集フィールドに直接入力します。
- 編集フィールドをクリックして、対象の値(PI AF 属性名、タイムスタンプ、表示セルの場所など)を含むセルまたはセル範囲をクリックします。
- 編集フィールドをクリックして、選択可能な有効な値のリストを表示します([ルートパス]および[データベース]フィールドでのみ使用可能)。
-  をクリックして検索ツールを開きます。それから、接続している PI Data Archive Server や PI AF Server を検索し、データアイテムを選択します。

リスト ▾がある場合、以下の作業を行えます：

- 計算やサンプリング方法などを、リストから選択できます。
-  をクリックして、対象の値(モード、時間単位、境界タイプ、プロパティなど)を含むセルまたはセル範囲をクリックできます。

注: 編集フィールドに文字列を入力するときは、引用符を使用しないでください。ただし、セルの場合、先頭に引用符(')を付けることで、Excel は入力値を文字列と見なします。

たとえば、[データアイテム]フィールドに PI ポイントを指定するには：

- ポイント名を表す文字列を入力します。
-  をクリックし検索ツールを開きます。PI Data Archive Server のポイントを検索します。
- ポイント名を含むセルの参照を入力します。
 - [データアイテム]フィールドをクリックします。
 - ワークシート上でセルをクリックします。

PI DataLink は、編集フィールドへセル参照を自動的に入力します。

時間入力

多くの PI DataLink 関数では、特定期間の値を取得するために、[Start time] と [End time] を入力する必要があります。その他の PI DataLink 関数では、特定時間の値を取得するために、[Time stamp] を入力する必要があります。時間を入力する際は、次のガイドラインに従ってください。

- 有効な PI 時間式を入力します。表記には、絶対時刻、相対時間の省略形、時間オフセットを含めることができます。常に特定の日付に関するデータを取得するには、10-Dec-16 19:12 などの絶対時刻を使用し、現在の時間を基準にした相対時間のデータを取得するには、t や -3h などの相対時間の省略形、時間オフセットを使用します。
- [End time] の入力にオフセット時間のみを入力すると、[Start time] の入力に対する相対的な時間を指定できます。
- [Start time] または [Time stamp] の入力にオフセット時間のみを入力すると、現在の時刻に対する相対時間を指定できます。
- 開始時刻に終了時刻よりも後の時刻を設定した場合、PI DataLink は結果を降順の時系列で表示します。
- セルに入力する時間式を一重引用符で始めると、文字列であることが示されます (例: '10-Dec-99 19:12 または '-3h)。

- セル参照には、Excel の絶対時間書式も使用できます(例 : 39299.6146。これは、8/5/2007 2:45:00 PM と同じです)。これは、Excel のタイムスタンプ表示書式で、1900 年から何日経っているかを表します。Excel では、同じタイムスタンプを、セルに割り当てられたどのような日時書式でも表示できます。
- また、「9:45」のような入力値を有効な時間値と解釈することも、ワークシートの有効な行範囲と解釈することも可能です。タスクペインのフィールドで、入力値の先頭に一重引用符を追加する(例: '9:45) と、PI DataLink はこれを時間として解釈します。

PI DataLink 関数には、[Time interval] の入力が必要なことがあります。時間間隔は 1 つの値で指定します。

- たとえば、「1d」や「30m」のように、値と時間単位を入力します。ただし、基準時間は入力しないでください。たとえば、32 分の間隔を指定するには、「32m」という文字列を直接入力するか、そのように入力されたセルを参照します。
- 周波数で間隔を入力する場合は、その周波数に相当する秒数に変換します。たとえば、周波数が 25Hz の場合、間隔は「0.04s」(=1/25 秒)と入力します。

注: PI DataLink は、Excel でサポートされている、1900 年からの日時計算システム(既定値)のみサポートしています。PI DataLink は、Excel の 1904 年からの日時計算システムはサポートしていないため、このシステムを使用した場合、誤ったタイムスタンプが返されます。

次の

[タイムスタンプ指定](#)

[時間間隔の仕様](#)

追加データ

PI DataLink は、取得した値に関する情報を追加できます。こうした追加データは、取得した値にコンテキストを提供します。追加データには、以下のものがあります:

- 値がいつ記録されたかを示すタイムスタンプ
- 時間間隔の開始時刻と終了時刻を示すタイムスタンプ
- 最大値と最小値が発生したタイムスタンプ
- サンプリング間隔の有効データ割合
- 値属性
- 手入力された注釈
- 検索元の PI Server 名

PI DataLink は、指定した追加データを、関数が返した値の隣の列(または行)に表示します:

- 時間データは、返却値の左の列(あるいは上の行)に表示されます。
- 他の関連データは、返却値の右の列(あるいは下の行)に表示されます。

表示書式

[設定 ウィンドウ] で、PI DataLink が関数配列で使用する時間や数値のデータ書式を指定します。2 つの設定があります:

- **数値書式**

関数が output する数値の書式。既定の設定である **General** は、Excel の[セルの書式設定]ウィンドウの**[標準]**に合わせて数値の書式(とタイムスタンプ以外のすべてのデータ)を設定します。

- **時間書式**

関数が output するタイムスタンプの書式。既定の時間書式は、標準 PI 時間書式 `dd-mmm-yy hh:mm:ss`, で、PI タイムスタンプの標準的な書式と同じです。ミリ秒タイムスタンプを表示するには、書式文字列(`.000`)の末尾に `dd-mmm-yy hh:mm:ss.000` を追加します。Excel は、マイクロ秒精度書式はサポートしていないため、注意してください。

Excel の[セルの書式設定]ダイアログボックスから有効な Excel 書式コードを使用して、書式文字列をカスタマイズできます。

[設定] ウィンドウの詳細については、「[Excel の PI DataLink 設定を管理する](#)」を参照してください。

さらに、[セルの書式設定] ウィンドウから、任意のセル(PI DataLink 関数を指定したセルを含む)に対し個々の数値と時間の書式を適用できます。数値と時間の書式を設定する方法についての詳細は、Excel のマニュアルを参照してください。PI DataLink は個々のセルの書式を次のように上書きまたは適用します。

- タスクペインで**[OK]**または**[適用]**をクリックした後で関数配列を書き込んだ場合、PI DataLink は、関数配列のセルに適用されている個々の数値または時間の書式を、[設定] ウィンドウの設定で上書きします。
- 関数配列を右クリックして**[DataLink 関数の再計算 (サイズ変更)]**をクリックした後で関数配列を書き込んだ場合、PI DataLink は、数値の書式を持つ配列内の左上のセルを探し、その書式(個々の書式または[設定] ウィンドウの書式)を、数値の書式を持つすべての配列セルに適用します。同様に、PI DataLink は時間の書式を持つ配列内の左上のセルを探し、その書式(個々の書式または[設定] ウィンドウの書式)を、時間の書式を持つすべての配列セルに適用します。

内挿値

多くの PI DataLink 関数は、内挿値を返します。これは、PI ポイントがタイムスタンプに関連付けられている場合に役立ちます。たとえば、タイムスタンプ以前で最後に記録された値ではなく、特定のタイムスタンプの内挿値を返すヒストリ値関数をリクエストすることもできます。内挿値は、記録されている値のタイムスタンプによって決定します。また、PI ポイントのタイプによって決定する場合もあります。

- タイムスタンプが最初に記録された値以前の場合は、関数は **Pt Created** または **No Data** を返します。
- タイムスタンプが 2 つの記録された値の間の場合、この関数では、記録されている値間を線形保管して値を決定します。デジタルステートポイントやステップポイントなどの離散値を保存しているポイントの場合、この関数は、タイムスタンプ以前で最後に記録された値を返します。
- 最後に記録された値よりタイムスタンプが後の場合、ポイントタイプによって返される値が次のように異なります。
 - 履歴データ用 PI タグの場合、この関数は、直近の値を返します。
 - の PI ポイントの場合、この関数は **No Data** を返します。

検索

PI DataLink は、2 つのデータアイテムの検索方法を提供します:

- **検索ツール**

検索ツールにより、テキスト検索や、PI Data Archive または PI AF データベースを参照することで、PI ポイントまたは PI AF 属性を検索できます。ツールを開く方法に応じて、検索ツールは検索したポイントや属性を、ワークシートまたは関数タスクペインに表示します。

- **アセットフィルター検索関数**

アセットフィルター検索関数を使用することで、検索条件に一致した PI AF エレメントを検索し、これらのエレメントを属性値に基づいて除外できます。この関数は、除外したエレメントまたは除外したエレメントの選択した属性を返すことができます。また、除外したアセットを、値または関数配列としてワークシートに貼り付けることができます。

データアイテムの検索

検索ツールを使用して、指定した PI Data Archive Server または PI AF Server のデータアイテムを検索します。ワークシートにデータアイテムを挿入し、後から PI DataLink 関数で参照できます。関数タスクペインに直接データアイテムを挿入することもできます。

1. 検索する PI Data Archive Server または PI AF Server に接続します。「[サーバー接続の管理](#)」を参照してください。
2. 検索ツールを開きます。
 - ワークシートにアイテムを挿入するには、データアイテムの挿入先セル範囲で左上隅のセルを選択し、[PI DataLink] タブの [Search] グループで [Search] をクリックします。
 - 関数タスクペインにアイテムを挿入するには、[データアイテム] フィールドの隣にある  をクリックします。
3. 検索範囲(パスの一番上に表示されます)を指定します。

初めて使用する場合、ツールはホームノードで始まります。ホームノードには、Connection Manager のリストにあるすべての PI Data Archive Server ( と表示) と PI AF Server ( と表示) が表示されます。検索範囲を 1 台の PI Data Archive Server や PI AF Server に限定する必要があります。さらに、検索範囲を PI AF Server 上の 1 つのデータベースに限定でき、ここから特定のエレメントや親属性に絞り込むことができます。

2 回目以降の使用時には、ツールでは、最後に検索した PI Data Archive Server、PI AF Server、または PI AF データベースに検索範囲が設定されたセッションで始まります。

次の操作を実行できます。

- [ブラウズペインを使った検索の絞り込み](#)
- [検索範囲パスを使った検索の絞り込み](#)

検索範囲にエレメントや属性を設定した場合、結果ペインに対象のエレメントや属性の直接の子属性が一覧表示されます。

4. 必要に応じて、範囲内のデータアイテムを検索し、結果ペインに一覧表示します。

a. ツールの一番上のフィールドに、選択した範囲内でデータアイテムを特定するための文字列を入力します。

ワイルドカード文字を使用して検索対象を増やすことができます。例：

- ・「sinusoid」を入力して、すべてのデータアイテム `sinusoid` を検索します。
- ・「sinusoid*」と入力すると、`sinusoidu` や `sinusoidu` など"sinusoid"で始まるすべてのデータアイテムが検索されます。
- ・「*u」と入力すると、`sinusoidu` など"u"で終わるすべてのデータアイテムが検索されます。
- ・すべてのデータアイテムを検索するには、「*」を入力します。

b. 検索範囲が PI Data Archive Server の場合、[フィルター]でリストとフィールドを設定し、取得された PI ポイントに対して追加の検索条件を指定します。

ペインの右上にある[クイックフィルター]リストには、事前定義された一般的な検索が組み込まれています。事前定義された検索を選択すると、検索ツールにより検索するリストとフィールドに自動的に値が入力されます。

このペインには、システムが割り当てる内蔵のポイント属性、`base` および `classic` ポイントクラスの属性、ポイント値、タイムスタンプ、ステータス (`Is good`) の仮想属性を含む、6 つのリストが表示されます。リストから属性を選択し、その属性の条件を指定します。ワイルドカード文字を入力することもできます。この検索では、暗黙的なワイルドカード文字は追加されません。最終的なリストには、ポイントタイプが含まれます。検索する格納済みの値の種類を選択します。

たとえば、[Descriptor]を選択し、「*vapor*」と入力すると、単語 `vapor` を `descriptor` ポイント属性のいずれかの場所に含む PI ポイントのみが返されます。

PI ポイント属性について詳しくは、PI Server のトピック「[PI ポイントクラスと属性](#)」を参照してください。

c. 検索を開始するには[検索] をクリックします。

検索ツールは、入力した文字列と一致するデータアイテムを指定範囲から検索し、検索したデータアイテムを結果ペインに返します。検索ツールは検索範囲下の全階層を検索します(これに対し、階層を参照した場合、結果ペインには検索範囲の直下の子属性のみが表示されます)。ツールは以下のものを検索します。

- ・名前が検索条件と一致した PI ポイント。
- ・名前が検索条件と一致した PI AF 属性。
- ・説明が検索条件と一致した PI AF 属性(PI AF Server 2015 以降を使用している場合)
- ・名前、説明、カテゴリ、またはテンプレートが検索条件と一致する親エレメントを持つ PI AF 属性。

注: PI Data Archive の検索で返すことができる PI ポイント数は、100,000 個までです。結果がすべて表示されるようにするために、検索結果がこの上限値を超えないように検索条件を指定してください。

5. 必要に応じて、結果ペインに表示される列を変更します。

- ・結果ペインを右クリックします。次に列名をクリックして、表示列に追加、または表示列から削除します。

- 列ヘッダーをドラッグして、表示列の順序を変更します。

6. 必要に応じて、[データアイテムの長さ]スライダーを動かし、データアイテムのパスを[ルートパス]と[データアイテム]列に分割します。

- [フルパス]を選択して、[データアイテム]列にフルパスを指定します。
- [名前のみ]を選択して、属性名またはポイント名のみを[ルートパス]列に、残りのパスを[ルートパス]列に指定します。
- 他の方法で列の内容を分割したい場合は、中間の位置を選択します。

上記の列の内容により、タスクペインのフィールドやワークシートの列に追加される内容が決定されます。

7. PI AF 属性をワークシートに挿入して、データアイテムのコンテンツを分割する場合、[ルートパス]コンテンツをワークシートに挿入する方法を指定します。

- [ドロップダウンリスト]を選択すると、([ルートパス]列の)パスと([データアイテム]列の)データアイテムを合わせたドロップダウンリストを、指定したワークシートのセルに挿入します。
挿入したドロップダウンリストとデータアイテムを PI DataLink 関数で参照していた場合、ユーザーがリストで違うパスを選択すると、ワークシートは取得データを動的に更新します。
注: PI DataLink は、ルートパスのリストをシートの ALL 列に挿入します。EZ 列にデータが存在する場合、右側の使用可能な列にパスを挿入します。
- [Column or row]を選択して、[Root path]列や[Data item]列の内容を指定したワークシートのセルに挿入します。
この場合、PI DataLink は常に[ルートパス]の内容をワークシートの 1 列目に挿入し、[データアイテム]の内容をワークシートの 2 列目に挿入します。

8. 挿入したいアイテムを選択して、[OK]をクリックします。

次の

[資産相対画面の作成](#)

ブラウズペインを使った検索の絞り込み

ブラウズペインは検索ツールの左側にあります。ブラウズペインには、現在の検索範囲の内容が表示されます(画面の一番上には、対応する検索範囲パスが表示されます)。検索範囲が PI AF Server にある場合、検索範囲より下にあり、かつ属性を含むデータベース、エレメント、属性のみがブラウズペインに表示されます。したがって、ブラウズペインには、子属性を持たないエレメントや属性は表示されません。

ブラウズペインを使用して、(検索範囲パスの設定と)検索範囲の限定を行います。たとえば、検索の範囲を特定の PI AF Server に限定し、次にそのサーバーの特定のデータベースに、さらにそのデータベースの特定のエレメントに絞り込むことができます。ブラウズペインの PI AF Server 階層で操作を行うと、現在選択中の検索範囲パスの直下にある属性が、結果ペインに表示されます。

ブラウズペイン上の作業:

- サーバーをクリックして、そのサーバーに検索を限定します。

ウィンドウの一番上にある検索範囲パスが更新され、選択したサーバーの内容でブラウズペインが更新されます。PI AF Server(で示されます)をクリックすると、ブラウズ

ペインはサーバー上のすべてのデータベースを表示します。PI Data Archive Server(データベース)で示されます)をクリックすると、ブラウズペインに、取得したPIポイントを属性値に基づいて絞り込むために使用できるフィルターのリストが表示されます。

- データベース(データベースで示されます)をクリックして、そのデータベースに検索を限定します。
選択したデータベースの最上位エレメントがブラウズペインにすべて表示されます。ウィンドウの一番上にある検索範囲パスが更新され、選択したデータベースの内容が表示されます。
- エレメント(エレメントで示されます)をクリックして、そのエレメントに検索を限定します。
選択したエレメントの下にあるすべての親エレメントと親属性がブラウズペインに表示されます。ウィンドウの一番上にある検索範囲パスが更新され、選択したエレメントの内容が表示されます。結果ペインに、選択したエレメントの属性がすべて表示されます。
- ブラウズペインは親属性のみを表示するのに対し、結果ペインは、選択したエレメントの下にある親属性と親ではない属性の両方を表示します。
- 属性(属性で示されます)をクリックして、その属性に検索を限定します。
選択した属性の下にある親属性がすべてブラウズペインに表示されます。ウィンドウの一番上にある検索範囲パスが更新され、選択した属性の内容が表示されます。結果ペインに、選択した属性の下にある属性がすべて表示されます。

検索範囲パスを使った検索の絞り込み

検索範囲パスは、検索ツールの一番上にあります。



検索範囲パスは、検索ツールがデータアイテムを検索する場所を示します。検索範囲パスを使用して、検索範囲を限定できます。たとえば、検索の範囲を特定の PI AF Server に限定し、次にそのサーバーの特定のデータベースに、さらにそのデータベースの特定のエレメントに絞り込むことができます。検索範囲パスの PI AF Server 階層で操作を行うと、現在選択中の検索範囲パスの直下にある属性が、結果ペインに表示されます。

検索範囲パスでの操作:

- [ホーム]をクリックして、ホームノードに移動します。ホームノードでは、接続管理に登録されたすべての PI Data Archive Server と PI AF Server がブラウズペインに表示されます。ホームノードでは、検索は行えません。最も広範な検索範囲は、単一の PI Data Archive Server または PI AF Server です。
- [ホーム]の隣の矢印をクリックして、次に特定のサーバーをクリックすると、そのサーバーに検索範囲が設定されます。
- サーバーをクリックすると、そのサーバー全体に検索範囲が設定されます。
- PI AF Server の隣の矢印をクリックして、次に特定のデータベースをクリックすると、そのデータベースに検索範囲が設定されます。
- データベースをクリックすると、そのデータベース全体に検索範囲が設定されます。
- データベースの隣の矢印をクリックして、次に特定のエレメントをクリックすると、そのエレメントに検索範囲が設定されます。

- エレメントをクリックすると、そのエレメント全体に検索範囲が設定されます。
- エレメントの隣の矢印をクリックして、次にサブエレメントや属性をクリックすると、そのサブエレメントや属性に検索範囲が設定されます。
- 親属性をクリックすると、そのすべてのサブ属性に検索範囲が設定されます。
- 親属性の隣の矢印をクリックして、次に特定のサブ属性をクリックすると、そのサブ属性が検索範囲として設定されます。

フィルタリングによるアセットの検索

アセットフィルター検索関数を使用することにより、PI AF データベースでエレメントを検索し、結果を属性値でフィルターできます。オプションで、フィルターされたエレメントを出力するか、またはフィルターされたエレメントの特定の属性を出力するかを選択できます。また、出力をワークシートに貼り付ける方法として、固定値または関数配列を選択できます。関数配列として貼り付けた場合、データは自動的に更新されます。

1. ワークシートで、取得したアセットを挿入する範囲の左上隅のセルを選択します。
2. [PI DataLink] タブの [検索] グループで、[アセットフィルター] をクリックして、[アセットフィルター検索] タスクペインを開きます。
3. [ルートパス] フィールドで、検索するエレメントへの共通パスを入力します。

共通パスは、サーバーとデータベースを含んでいる必要があります。オプションで任意の親エレメントを含むことができます。 *ServerName**DatabaseName**ParentElementName* の書式で指定します。

たとえば、サーバー MyDatabase のデータベース MyPIAFServer のルートレベルでエレメントを検索するには、「\\MyPIAFServer\MyDatabase」と入力します。同じデータベースの Boilers エレメントの下のエレメントを検索するには、「\\MyPIAFServer\MyDatabase\Boilers」と入力します。

4. 取得する PI AF エレメントを指定します。
 - a. [エレメントテンプレート] リストから、取得したエレメントのテンプレートを選択します。

属性値に基づいてエレメントをフィルターするために、テンプレートを選択する必要があります。ベーステンプレートを選択すると、派生したテンプレートからのエレメントが取得されることに注意してください。
 - b. [エレメント名] フィールドで、取得するエレメントの名前を入力します。

ワイルドカード文字を使用して、名前の一部を指定することもできます。
 - c. [エレメントカテゴリ] リストから、取得するエレメントのカテゴリを選択します。
 - d. [エレメントの説明] フィールドで、取得するすべてのエレメントの説明に含まれるテキストを入力します。

ワイルドカード文字を使用して、説明の一部を指定することもできます。
 - e. [ルートパス] フィールドで指定したレベルのエレメントのみが取得されるようにするには、[ルートレベルに限定] チェックボックスを選択します。子エレメントも同様に取得するには、このチェックボックスをオフにします。
 - f. オプション: [属性値フィルター] 表で、取得するエレメントの属性値に対する条件を指定します。

属性値条件を指定する前に、エレメントテンプレートを選択する必要があります。選択したエレメントテンプレート、または選択したエレメントテンプレートの基本テンプレートで定義されている属性の値のみを使用してフィルターできます。

最大 5 つの条件を指定できます。条件ごとに、3 つのフィールドを設定します。

- **[属性]** リストから、選択したエレメントテンプレートに基づいて、エレメントにある属性を選択します。
- **[演算子]** リストから、**[=]**、**[<]**、**[>]**などの関係演算子を選択します。文字列、ブール型、または列挙型の値を格納する属性については、有効な演算子は**[=]**と**[<>]**のみです。
- **[値]** フィールドに、フィルターで使用する値を入力します。文字列属性については、ワイルドカード文字を使用できます。

たとえば、**Manufacturer** で始まる ABC 属性と 94102 から 94188 までの ZipCode 属性を持つエレメントを取得するには、次の 3 つの条件を入力します。

Manufacturer = ABC*

ZipCode >= 94102

ZipCode <= 94188

最高のパフォーマンスを達成するために、値が PI AF データベースに格納されている属性のみに制限します（つまり、データ参照を持つ属性を除外します）。データ参照を持つ属性を指定すると、関数は最大フィルター検索カウント用の設定を使用して、一致する属性値を検索するエレメントの数を制限します。「[Excel の PI DataLink 設定を管理する](#)」を参照してください。

5. ワークシート出力を指定します。

- **[表示属性]** リストから、表示する属性を選択し、表示する順序を設定します。

属性が何も選択されていない場合、アセットフィルター検索関数は一致するエレメントのみを返します。1 つ以上の属性が選択されている場合、関数はエレメントごとに選択された属性を返します。

既定では、リストには選択されたエレメントテンプレートの属性が含まれます。次の操作を実行できます。

- **[すべて選択]** チェックボックスを選択して、リストされているすべての属性を含めます。
- 属性に対応するチェックボックスを選択して、その属性を含めます。または、チェックボックスをオフにして、その属性を除外します。
- リストの下部にある空白のチェックボックスの隣に、属性の名前を入力します。

- 属性を選択し、 をクリックして、表示される属性のリストで、その属性を上位に移動します。
- 属性を選択し、 をクリックして、表示される属性のリストで、その属性を下位に移動します。
- 属性を選択し、 をクリックして、表示される属性のリストから、その属性を削除します。

- 一致するエレメントまたは属性をワークシートに貼り付ける方法を示します。

- [列]をクリックして、指定したワークシートのセルにエレメントまたは属性へのフルパスを挿入します。

- [ドロップダウンリスト]をクリックして、属性へのパス(つまり、固有のサーバー、データベース、および親エレメント)と、その後に選択した属性を含むドロップダウンリストを、指定したワークシートのセルに挿入します。

1つ以上の属性を選択している場合のみ、[ドロップダウンリスト]を選択します。

別の PI DataLink 関数から、挿入したドロップダウンリストを[ルートパス]フィールドから、属性を[データアイテム]フィールドから参照して、資産相対画面を作成できます。「[資産相対画面](#)」を参照してください。

c. 出力書式を示します。

- [関数配列]をクリックして、関数配列を貼り付けます。これは、他の PI DataLink 関数の出力書式です。関数配列を使用すると、容易に入力をタスクペインから更新でき、出力を再計算できます。

- [値]をクリックして、出力を値として貼り付けます。容易に値をコピーして、他の場所で使用できます。

注: 関数配列は、特に関数の結果が頻繁に変化する場合に役立ちます。ただし、関数配列の再計算は、ワークシートを開くごとに実行されるので、開く操作が遅くなることがあります。これに比べて、結果が変化しないことが想定され、関数配列の再計算が終了するのを待機したくない場合、値が便利です。

d. [表示セル]に、返されたアセットを挿入したいワークシートの左上隅のセルが含まれていることを確認します。

6. [OK]をクリックして、一致するエレメントまたは属性をワークシートに挿入し、タスクペインを閉じます。

PI DataLink 設定

時間と数値の書式、返されるタイムゾーン、関数から返される文字列、タスクペインが開くタイミング、検索結果のペースト方法など、多くの PI DataLink の動作が設定により制御されます。コンピューター上のユーザー各自が個別に設定を保存しています。ユーザーは Excel から直接これらの設定を表示、変更できます。

PI DataLink では、各ユーザーの **AppData** ディレクトリに保存された **OSisoft.PIDataLink.xml** ファイルから設定を取得します。

%UserProfile%\AppData\Local\OSisoft,_Inc\PIDataLink

管理者は、特定のコンピューター上のすべてのユーザーについて設定を変更したり、特定のコンピューター上のすべてのユーザーから設定を削除したりすることができます。

Excel の PI DataLink 設定を管理する

[設定]ダイアログボックスを使用して、PI DataLink の全般設定を行い、PI DataLink 関数用の既定の出力書式を指定します。設定は、特定コンピューターの特定ユーザーにのみ適用されます。このウィンドウから、PI ポイントの構成データや、PI AF の非時間ベースのデータのキャッシュもクリアできます。

1. [PI DataLink]タブの[Resources]グループで、[Settings]をクリックします。

2. PI DataLink のキャッシングを手動でクリアするには、[キャッシングのクリア]をクリックします。

既定では、PI DataLink は PI ポイントの構成データ、および PI AF の非時間ベースのデータ(PI ポイントデータ参照およびイベント以外のデータ)をキャッシングします。このキャッシングにより、パフォーマンスが改善します。関数を計算する前に、PI DataLink で、キャッシングが最後に消去された時刻をチェックします。キャッシングが過去 6 か月以内に消去されていない場合は、自動的にキャッシングが消去されます。キャッシングをクリアして、更新されたデータを素早く取得するには、手動でキャッシングをクリアします。

3. 設定を更新するには、希望する設定を行い、[OK]をクリックします。

設定	説明
ブランクの代わりに #N/A を表示	このチェックボックスを選択すると、再計算で取得した値の数が、関数配列の表示可能数より少ない場合、値がないセルには、空白ではなく「#N/A(該当なし)」と表示されます。 このオプションは、Excel のグラフ作成機能で関数の結果をプロットしている場合に特に役立ちます。
ロケールの影響を受けない	このチェックボックスを選択すると、PI DataLink はクライアントマシンのロケール設定に関係なく、PI 時間書式の規則に従って入力時間文字列を解釈します。PI 時間は、すべての文字列が英語表記であるものとし、dd-mmm-yyyy hh:mm:ss の日時書式を使用します。 このチェックボックスをオフにすると、クライアントマシンのロケール設定で日時書式が解釈され、必要に応じて PI 時間書式の規則に戻ります。
クリックによる作業ウィンドウの自動表示を無効にする	関数セルのクリック時に関数タスクペインを自動的に開く機能を無効にしたい場合に選択します。
「サイズ変更してすべて表示」のメッセージを無効にする	このチェックボックスを選択すると、再計算時に関数配列の値の個数が、表示できる個数を超えて、警告メッセージが表示されなくなります。
クライアントタイムゾーン	このチェックボックスを選択すると、クライアントマシンのタイムゾーンで入力タイムスタンプが解釈され、出力タイムスタンプが表示されます。一部の関数では、特定のタイムスタンプが PI Data Archive Server のタイムゾーンで表示されます。これらの例外については、 タイムゾーン設定の制限 を参照してください。

設定	説明
PI Data Archive のタイムゾーン	<p>このチェックボックスを選択すると、PI Data Archive Server のタイムゾーンで入力タイムスタンプが解釈され、出力タイムスタンプが表示されます。この設定は、PI ポイントデータ参照である PI ポイントまたは PI AF 属性のいずれかを含むデータアイテムまたは式に対してのみ適用できます。データアイテムや式が PI ポイントデータ参照ではない PI AF 属性を含む場合には、PI DataLink はクライアントマシンのタイムゾーンを使用して入力および出力のタイムスタンプを解釈します。</p> <p>このオプションが関係するのは、PI Data Archive Server のタイムゾーン設定がクライアントマシンのタイムゾーン設定と異なる場合に限られます。</p>
UTC タイムゾーン	<p>このチェックボックスを選択すると、協定世界時(UTC)で入力タイムスタンプが解釈され、出力タイムスタンプが表示されます。一部の関数では、この設定はサポートされていません。これらの例外の詳細については、タイムゾーン設定の制限を参照してください。</p>
行内	選択すると、検索の結果をワークシート上の 1 行に貼り付けます。
列内	選択すると、検索の結果をワークシート上の 1 列に貼り付けます。既定値。
数値書式	<p>関数が出力する数値の書式を指定します。書式文字列には、[Excel Format] ウィンドウ(表示書式を参照)上で有効な任意の数値書式コードを指定できます。</p> <p>PI DataLink は、インストール時に Excel の既定の数値書式をこのフィールドに指定します。ただし、それ以前に対象コンピューターの DataLink でユーザーが独自の書式を設定している場合は、ユーザーの設定が有効になります。Excel のロケールが異なる場合、この書式にはそのロケールに対応する構文が含まれます。例：</p> <p>35.03</p> <p>は、フランスの Excel では、</p> <p>35,03</p> <p>と表示されます。</p>

設定	説明
時間書式	<p>関数が出力するタイムスタンプの書式を指定します。時間文字列には、[Excel Format] ウィンドウ上で有効な任意の時間書式コードを指定できます。</p> <p>PI DataLink は、インストール時に標準の PI タイムスタンプ書式をこのフィールドに指定します。ただし、それ以前に対象コンピューターの DataLink でユーザーが独自の書式を設定している場合は、ユーザーの設定が有効になります。Excel のロケールが異なる場合、この書式にはそのロケールに対応する構文が含まれます。</p> <p>例：</p> <p>dd-mmm-yy hh:mm:ss</p> <p>は、フランスの Excel では、</p> <p>jj-mmm-aa hh:mm:ss</p> <p>と表示されます。</p>
最大イベントカウント	Explore Events 関数および Compare Events 関数がプレビューまたはワークシートに返すイベントの最大数を指定します。
最大フィルター検索カウント	Asset Filter Search 関数がワークシートに返すエレメントの最大数を指定します。ただし、データ参照のある属性に対して属性値フィルターを指定している場合を除きます。この場合、データ参照のある属性を検索する必要がある要素の総数を指定します。この数をあまりに低く設定すると、想定よりも少ない結果が返されることがあります。
計算 (F9)	選択すると、自動更新機能による再計算時にすべての揮発性関数(と揮発性関数を参照する関数)が再計算されます。現在値関数は、PI DataLink 関数で唯一の揮発性関数です。
全計算 (Ctrl+Alt+Shift+F9)	選択すると、自動更新機能による再計算時に、揮発性にかかわらず、すべての関数が再計算されます。
間隔 (秒)	自動更新機能による再計算の間隔を秒数で入力します。最小値は 5 秒です。「0」を入力すると、前回の計算にかかった時間に基づいて、PI DataLink が自動更新間隔を計算します。

タイムゾーン設定の制限

タイムゾーン設定には、特定の制限が適用されます：

- 注釈にタイムスタンプが含まれている場合、圧縮データ関数は、そのタイムスタンプを常に PI Data Archive Server のタイムゾーンで表示します。
- プロパティ関数は、常に特定のタイムスタンプ(作成日および変更日など)を含む PI ポイント属性の値を PI Data Archive Server のタイムゾーンで表示します。

コンピューター上のすべてのユーザーについて PI DataLink の設定を変更する

管理者はバッチファイルを使用して、コンピューター上のすべてのユーザーについて PI DataLink の設定を変更できます。

- すべてのユーザーの希望する設定を含む **OSIsoft.PIDataLink.xml** ファイルを作成します。

ユーザー アカウントのフォルダー（%UserProfile%\AppData\Local\OSIsoft_Inc\PIDataLink）からファイルをコピーするか、からファイルをコピーするか、次のデフォルトの内容で始まるファイルを作成できます。

```
<Settings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <CTPDockWidth>200</CTPDockWidth>
  <CTPFloatLeft>-1</CTPFloatLeft>
  <CTPFloatTop>-1</CTPFloatTop>
  <CTPFloatWidth>200</CTPFloatWidth>
  <CTPFloatHeight>-1</CTPFloatHeight>
  <CTPDockPosition>right</CTPDockPosition>
  <CopyItemsInRow>0</CopyItemsInRow>
  <CopyServerName>0</CopyServerName>
  <UseServerTime>0</UseServerTime>
  <DisplayEndTime>0</DisplayEndTime>
  <DisplayNA>0</DisplayNA>
  <LocaleIndependent>0</LocaleIndependent>
  <DisableAutoReinit>0</DisableAutoReinit>
  <DisableResizeMessage>0</DisableResizeMessage>
  <NFormat>General</NFormat>
  <TFormat>dd-mmm-yy hh:mm:ss</TFormat>
  <AutoUpdateCalculateMode>0</AutoUpdateCalculateMode>
  <AutoUpdateInterval>0</AutoUpdateInterval>
  <LastSearchFullPath>false</LastSearchFullPath>
  <LastSearchPath> </LastSearchPath>
  <LastEFDatabase> </LastEFDatabase>
  <MaxEFCount>1000</MaxEFCount>
  <MaxAFSearchCount>10000</MaxAFSearchCount>
</Settings>
```

- 希望する設定のファイルをすべてのユーザーに割り当てるバッチファイルを作成します。

たとえば、**deploySettings.bat** というバッチファイルは次の要領で作成できます。

```
@echo off

IF NOT EXIST "%CD%\OSIsoft.PIDataLink.xml" (
ECHO OSIsoft.PIDataLink.xml file not found in this directory.
EXIT /B 2
)

SET ProfileBase=%SystemDrive%\Users
SET AppDir=APPDATA\Local
SET DefaultUsername=Default

ECHO Deploying XML settings to all users.
for /f "tokens=*" %%a in ('dir /b /ad-h "%ProfileBase%"') do if 1==1 (
```

```
REM No need to copy to Public and All Users folders
ECHO %%a | findstr /i "all.users public" >nul 2> nul
if errorlevel 1 (
mkdir "%ProfileBase%\%%a\%AppDir%\OSIsoft,_Inc"
mkdir "%ProfileBase%\%%a\%AppDir%\OSIsoft,_Inc\PIDataLink"
xcopy /f /y "%CD%\OSIsoft.PIDataLink.xml" "%ProfileBase%\%%a\%AppDir%
\OSIsoft,_Inc\PIDataLink\
)

)

REM Apply to default user
mkdir "%ProfileBase%\%DefaultUsername%\%AppDir%\OSIsoft,_Inc"
mkdir "%ProfileBase%\%DefaultUsername%\%AppDir%\OSIsoft,_Inc\PIDataLink"
xcopy /f /y "%CD%\OSIsoft.PIDataLink.xml" "%ProfileBase%\%DefaultUsername%\%AppDir%
\OSIsoft,_Inc\PIDataLink\"
```

- 希望する設定を含む **OSIsoft.PIDataLink.xml** ファイルと同じディレクトリにこのバッチファイルを保存します。
- エクスプローラーでこのバッチファイルを右クリックし、**【管理者として実行】**をクリックします。スクリプトによってこのコンピューター上のすべてのユーザーの既存の設定が上書きされ、希望する設定に置き換わります。この操作を元に戻すことはできません。このコンピューター上の新規ユーザーにも、同じ設定が適用されます。
- このスクリプトを実行した後に作成する新規アカウントには、これらの優先設定を割り当てない場合、Default ユーザー アカウントから **OSIsoft.PIDataLink.xml** ファイルを削除します (**C:\Users\Default\AppData\Local\OSIsoft,_Inc\PIDataLink** にあります)。

コンピューター上のすべてのユーザーから PI DataLink の設定を削除する

管理者はコンピューター上のすべてのユーザーから PI DataLink の設定を削除できます。ユーザーが次に PI DataLink を開くと、新しくインストールされたデフォルト設定が有効になります。

- OSIsoft.PIDataLink.xml** ファイルを削除するバッチファイルを作成します。

```
@echo off

SET ProfileBase=%SystemDrive%\Users
SET AppDir=APPDATA\Local
SET DefaultUsername=Default

ECHO Deleting XML settings file from all users.
for /f "tokens=*" %%a in ('dir /b /ad-h "%ProfileBase%"') do if 1==1 (
ECHO Deleting "%ProfileBase%\%%a\%AppDir%\OSIsoft,_Inc".
rmdir "%ProfileBase%\%%a\%AppDir%\OSIsoft,_Inc" /s /q
)

ECHO Deleting "%ProfileBase%\%DefaultUsername%\%AppDir%\OSIsoft,_Inc".
rmdir "%ProfileBase%\%DefaultUsername%\%AppDir%\OSIsoft,_Inc" /s /q
```

- エクスプローラーでこのバッチファイルを右クリックし、**【管理者として実行】**をクリックします。

サーバー接続の管理

接続管理では、PI Data Archive Server や PI AF Server への接続の管理(既定の PI Data Archive Server や PI AF Server の選択を含む)を行います。

1. [PI DataLink]タブの[リソース]グループで[設定]をクリックして、[設定]ウィンドウを開きます。
2. [接続管理]をクリックして、[サーバー]ウィンドウを開きます。このウィンドウでは、定義したサーバーの接続、接続のステータス、既定サーバーが表示されます。
3. 必要に応じて、接続の修正を行います。
 - [Asset Server の追加]をクリックして[PI AF Server プロパティ]ウィンドウを開きます。このウィンドウでは、PI AF Server への接続を定義できます。
 - [Data Server の追加]をクリックして[PI Data Archive プロパティ]ウィンドウを開きます。このウィンドウでは、PI Data Archive Server への接続を定義できます。
 - 現在接続していないサーバーを選択し[接続]をクリックして、そのサーバーに接続します。

PI AF サーバーに接続するために、サーバーがOIDCをサポートするように構成されている場合、PI DataLink はログインユーザーの Windows 資格情報またはOpen ID Connect (OIDC) 資格情報を使用できます。PI Data Archive サーバーに接続するために、PI DataLink は、ログインユーザーの Windows 資格情報とOIDC 資格情報に加えて、PI Trust またはデフォルトユーザーを使用できます。

- 現時点で既定の接続ではないサーバーを選択して[既定として設定]をクリックすると、そのサーバーは既定の接続になります(PI Data Archive Server または PI AF Server)。
- サーバーを選択して[プロパティ]をクリックすると、そのサーバーの接続プロパティが表示されます。
- サーバーを右クリックして[削除]をクリックすると、そのサーバーへの接続定義が削除されます。

章 4

ワークシートの作成

このセクションでは、PI DataLink を使用してワークシートを作成するプロセス、および作成できる特定の画面のタイプについて説明します。画面のトピックには、これらの画面を作成する手順が含まれます。

ワークシートの作成プロセス

PI DataLink の基本ツールや考え方について理解できたら、PI DataLink 関数を使用してワークシートを作成しましょう。

目標の設定

以下の質問に答えてください:

- ワークシートに表示するデータはどのようなデータですか。パフォーマンス監視やビジネスの問題解決のためのデータですか。
- データはどこにありますか。
- どのようにして本質的なポイントを伝え、必要な背景を提示し、最も効果的に情報を表示できるでしょうか。

上記の質問に答えることによって、最も作業に適するデータを返す PI DataLink 関数はどれなのかを、判断しやすくなります。

アプローチの選択

次のような手法でワークシートを作成できます:

- 関数ベースのアプローチ

PI DataLink 関数をワークシートに追加し、必要に応じて関数の周囲に画面を構築します。たとえば、対象のデータが表すものを明確にするために、ポイント情報を追加します。まだ要件を調査している場合、PI DataLink の機能を習得したりしている場合に、このアプローチは有効です。

- 構成ベースのアプローチ

ワークシート作成時に構成を追加します。検索ツールを使用して、ワークシートの主題となるデータアイテムを挿入します。次に対応するデータを取得する関数を追加します。これには、PI DataLink の知識と構成計画が必要ですが、一度データアイテムをワークシートに配置すれば、セル参照を使用することで、配置したデータアイテムから関数を簡単に構成できます。この方法では、ワークシートの再利用が容易になります。

- レポートベースのアプローチ

2 つのワークシートを使用します。2 つ目のワークシートに PI DataLink 関数を配置します。1 つ目のワークシートで上記の関数の結果を参照します。Excel のセキュリティ機能を使用して、2 つ目のシートにある関数構文とビジネスロジックを保護します。シートの作成者が管理者である場合、または他のユーザーのためにワークシートを作成している場合、このアプローチは有効です。また、PI Data Link Server によって配布されたドキュメントにも有効です。

次の

[PI DataLink 関数](#)

[データアイテムの検索](#)

大規模データの取得

関数の各データアイテムで、PI DataLink は、PI Data Archive または PI Asset Framework のいずれかへの呼び出しを行い、値を取得する必要があります。PI DataLink で多数のデータアイテムの値を取得する場合、呼び出しごとに個別に処理するため、取得時間が長くなることがあります。この取得時間を短縮するため、PI DataLink は、特定の条件で特定の関数を一括して呼び出します。

一括呼び出しは、次の条件を満たした場合に行われます。

- 関数の入力でデータアイテムを指定し、式やフィルター式を指定しない場合
- 関数の入力で、セル範囲を参照してデータアイテムを指定する場合
- PI DataLink 設定で、クライアントのタイムゾーンでタイムスタンプを指定している場合

上記の条件を満たした場合、次の関数で一括呼び出しが行われます。

- 現在値
- ヒストリ値
- 演算データ(時間間隔の指定なし)

PI DataLink は、単一の関数配列で一括呼び出しから値を返します。

一括呼び出しを行うと、関数が 1000 個のデータアイテムを取得した場合、または PI DataLink とデータソース間のレイテンシーのある環境では、パフォーマンスが改善します。関数が 10000 個以上のデータアイテムの値を取得する場合は、一括呼び出しを行うことで、増大の命令により取得時間が短縮します。したがって、多数のデータアイテムの値を取得するワークシートを作成するときは、これらの関数を必須条件で使用することを検討してください。

資産相対画面

資産相対画面は、PI AF 属性とそれに対応するアセット(PI AF エレメント)の値を表示します。他のアセットを選択することで、そのアセットに対する値を表示できます。

[エレメント「React1」に対する資産相対画面](#)

\\DLA\\PI\\MyTest\\Reactors\\React1	▼
Manufacturer	ACME
Manufacturer Serial Number	A123456
Temperature Attribute	49.12171555

エレメント「React2」に対する資産相対画面

\\DLA\\PI\\MyTest\\Reactors\\React2	▼
Manufacturer	ACME
Manufacturer Serial Number	A123458
Temperature Attribute	153.0063477

資産相対画面は、複数のアセットで共通する属性名を必要とします。たとえば、同じエレメントと属性テンプレートに基づいた複数の PI AF エレメントを使用して、資産相対画面を作成できます。

資産相対画面の作成

資産相対画面を作成するには、ルートパスドロップダウンリストを使用して、共通構造を持つデータアイテムをワークシートに取り込みます。次に、そのセルを参照する PI DataLink 関数を追加します。資産相対画面を作成後に、リストから他のアセットを選択して、そのアセットの属性値を表示できます。

1. [共通の構造を持つデータアイテムのワークシートへの取り込み](#).
2. [挿入されたデータアイテムを参照する PI DataLink 関数の構成](#).

次の

[データアイテムの検索](#)

共通の構造を持つデータアイテムのワークシートへの取り込み

資産相対画面は、共通構造を持つデータアイテム(PI AF 属性)を含むワークシートと、これらのデータアイテムに対するルートパスドロップダウンリストを必要とします。

1. データアイテムの挿入先のワークシートセルを選択します。
2. [PI DataLink]タブで[検索]をクリックして、検索ツールを開きます。
3. 検索範囲を設定し、対象のデータアイテムを指定して、[検索]  をクリックします。

資産相対画面の場合、同じ属性構造および共通の親エレメントを持つ PI AF 属性を検索します。これらの属性は同じ属性テンプレートを持つこともあります。たとえば、施設にあるすべての化学反応装置を検索する場合もあります。

4. データアイテムのパスを分割して、画面に表示するアセットを[ルートパス]列に表示します。

リストされた属性の固有のエレメントが[ルートパス]列に表示されるまで、[データアイテムの長さ]のスライダーを動かします。通常、これは[名前のみ]か、それに近い設定になります。

ルートパス	データアイテム	説明
¥¥DLAFTP¥Doug¥Reactors¥React1	Manufacturer	
¥¥DLAFTP¥Doug¥Reactors¥React1	Manufacturer Last Maintained	
¥¥DLAFTP¥Doug¥Reactors¥React1	Manufacturer Serial Number	
¥¥DLAFTP¥Doug¥Reactors¥React1	Temperature Attribute	
¥¥DLAFTP¥Doug¥Reactors¥React10	Manufacturer	
¥¥DLAFTP¥Doug¥Reactors¥React10	Manufacturer Last Maintained	
¥¥DLAFTP¥Doug¥Reactors¥React10	Manufacturer Serial Number	
¥¥DLAFTP¥Doug¥Reactors¥React10	Temperature Attribute	
¥¥DLAFTP¥Doug¥Reactors¥React2	Manufacturer	
¥¥DLAFTP¥Doug¥Reactors¥React2	Manufacturer Last Maintained	
¥¥DLAFTP¥Doug¥Reactors¥React2	Manufacturer Serial Number	
¥¥DLAFTP¥Doug¥Reactors¥React2	Temperature Attribute	
¥¥DLAFTP¥Doug¥Reactors¥React3	Manufacturer	
¥¥DLAFTP¥Doug¥Reactors¥React3	Manufacturer Last Maintained	
¥¥DLAFTP¥Doug¥Reactors¥React3	Manufacturer Serial Number	

データアイテムの長さ: ルートパスの挿入: ドロップダウンリスト 列または行

フルパス: 名前のみ

- ワークシートに挿入したいアイテムを選択します。
- [ルートパスの挿入]の下にある[ドロップダウンリスト]を選択します。
- [OK]をクリックします。

PI DataLink は、固有のデータアイテムをワークシートのドロップダウンリスト(ドロップダウンリストにはルートパス列からユニークなアセットが含まれます)の配下に取り込みます。

\\DLAFTP\\MyTest\\Reactors\\React1

Manufacturer

Manufacturer|Last Maintained

Manufacturer|Serial Number

Temperature Attribute

注: PI DataLink は、ルートパスのリストをシートの ALL 列に挿入します。EZ 列にデータが存在する場合、右側の使用可能な列にパスを挿入します。

挿入されたデータアイテムを参照する PI DataLink 関数の構成

資産相対画面では、PI DataLink 関数を設定し、ワークシートに挿入したデータアイテムとルートパス(共通の構造を持ったデータアイテムとその構造に対するルートパス)に対するデータを取得する必要があります。

- 挿入したセル範囲にある最初のデータアイテムの隣のセルをクリックします。
- [PI DataLink]タブで、値取得に使用する関数をクリックします。

クリックしたセルを[表示セル]に指定した状態で、関数タスクペインが開きます。

3. 関数タスクペインで[データアイテム]フィールドをクリックして、今挿入したデータアイテムを含むセル範囲を選択します。

PI DataLink はセル範囲をフィールドに入力します。

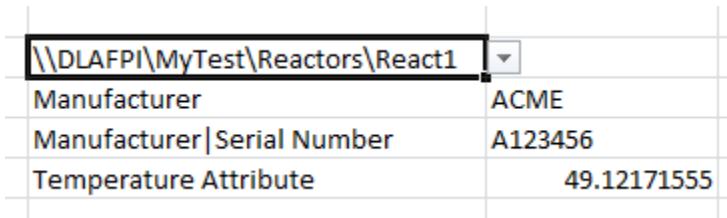
4. 関数タスクペインで[ルートパス]フィールドをクリックします。次に、ルートパスドロップダウンリストのあるセルを選択します。

PI DataLink はセル参照をフィールドに入力します。

5. 必要に応じて、他の関数の入力値を設定します。

6. [OK]をクリックして、関数配列をワークシートに挿入します。

リストから他のアセットを選択することで、そのアセットに対する値を表示できます。



\\DLA\\PI\\MyTest\\Reactors\\React1	ACME		
Manufacturer	ACME		
Manufacturer Serial Number	A123456		
Temperature Attribute	49.12171555		

リストから他のアセットを選択することで、そのアセットに対する値を表示できます。

ワークシートのイベント

PI DataLink を使用して、PI AF に格納されたイベントを表示、理解、および分析するための画面を作成できます。システム構成、および収集したデータタイプにより、格納するイベントが決定されます。たとえば、バッチプロセスに関する情報を格納するイベントまたは計算から作成されたイベントがシステムにある場合などです。

次の 2 つの PI DataLink 関数を使用してイベントを取得できます:

- イベントの検索関数

イベント階層にあるイベントの調査に役立ちます。イベントの検索関数は行ごとに 1 つのイベントを返します。イベントの検索関数を使用して、特定のエレメントのイベントを分析する場合に、単純な階層でデータを検証できます。たとえば、特定のボイラーの停止時間を検証できます。

エレメントに関連するイベントの調査



イベント名	開始時刻	終了時刻	プライマリ エレメント	ReasonCode	Shutdown
20140621 19:26	21-Jun-14 19:26:00	21-Jun-14 19:29:00	Boiler5	P	Planned
20140621 19:28	21-Jun-14 19:28:20	21-Jun-14 21:10:20	Boiler5	P	Planned
20140621 20:19	21-Jun-14 20:19:51	21-Jun-14 20:45:51	Boiler5	E	Emergency
20140621 20:26	21-Jun-14 20:26:00	21-Jun-14 20:29:00	Boiler5	P	Planned

イベントの検索関数を使用して、イベントのタイプに対する子イベントを分析する場合に、全階層内のイベントを検証することもできます(イベントテンプレートは、多くの場合、イベントのタイプを示します)。たとえば、タービンの起動フェーズを検証できます。

子イベントのあるイベントの調査

イベント名	Child1	イベント テンプレート	開始時刻	終了時刻	プライマリ エレメント
TurbineStartup.1.1		TurbineStartUp	21-Jun-14 19:28:20	21-Jun-14 21:10:20	Turbine1
TurbineStartup.1.1	Phase1	StartUpPhase1	21-Jun-14 19:28:20	21-Jun-14 20:12:20	Turbine1
TurbineStartup.1.1	Phase2	StartUpPhase2	21-Jun-14 20:12:20	21-Jun-14 20:39:50	Turbine1
TurbineStartup.1.1	Phase3	StartUpPhase3	21-Jun-14 20:39:50	21-Jun-14 21:10:20	Turbine1
TurbineStartup.3.1		TurbineStartUp	21-Jun-14 21:10:20	21-Jun-14 22:25:20	Turbine1
TurbineStartup.3.1	Phase1	StartUpPhase1	21-Jun-14 21:10:20	21-Jun-14 21:15:20	Turbine1
TurbineStartup.3.1	Phase2	StartUpPhase2	21-Jun-14 21:15:20	21-Jun-14 21:22:20	Turbine1
TurbineStartup.3.1	Phase3	StartUpPhase3	21-Jun-14 21:22:20	21-Jun-14 22:25:20	Turbine1

- イベントの比較関数

階層的なイベントの比較に役立ちます。イベントの比較関数は、関連するイベントの属性を单一の行に返すことができます。イベントの比較関数で作成した画面で、各行に取得したイベントとともに、そのイベントの子イベント、および親イベントの情報を表示できます。これらの情報を含めることで、取得したイベントを容易に比較できます。たとえば、起動イベントのフェーズイベントの情報を含めると、複数の起動イベントを比較できます。

イベントの比較関数はパスを使用して属性を識別します。このため、属性名、および階層の位置は画面に影響を与えます。イベントの比較関数の機能は、イベントおよびその属性の構造により異なります:

- 同一の名前を持つ子イベントのあるイベント

同一の名前の子イベントを持つイベントのセットがある場合、子イベントおよび親イベントの情報を含めることで、これらのイベントを比較できます。類似した名前の子イベントを持つイベントを検索するための条件を指定します。イベントの比較関数は、一致した各イベントと同じ行に子イベントの情報を含めることができます。たとえば、Phase1、Phase2 および Phase3 のような、類似した名前のフェーズイベントがあるタービンの起動イベントを比較できます。

子イベント情報を含めることによるイベントの比較

イベント名	イベント テンプレート	開始時刻	終了時刻	期間	Phase1 期間	Phase2 期間	Phase3 期間
TurbineStartup.3	TurbineStartUp	06-Jun-14 12:29:00	06-Jun-14 14:11:00	0 1:42:00	0 0:12:00	0	0
TurbineStartup.5	TurbineStartUp	08-Jun-14 10:21:00	08-Jun-14 10:42:00	0 0:21:00	0 0:07:00	0	0

イベントの比較関数は、一致したイベントと同じ行に親イベントの情報を含めることができます。親イベントからの情報を含めることで、一致したイベントの詳細な情報を提供できます。たとえば、一致したタービン起動イベントについて、親プロセスイベントの情報があれば、比較精度が向上する可能性があります。

子イベントおよび親イベント情報を含めることによるイベントの比較

イベント名	イベント テンプレート	開始時刻	イベント名	イベント テンプレート	開始時刻
Process.East.3	MachineCycle	06-Jun-14 12:29:38	TurbineStartup.3.3	TurbineStartUp	20-Jun-14 12:29:38
Process.South.3	MachineCycle	06-Jun-14 12:29:38	TurbineStartup.5.3	TurbineStartUp	20-Jun-14 12:29:38

- 異なる名前を持つ子イベントのあるイベント

異なる名前を持つ子イベントのあるイベントのセットがある場合、親イベント情報を含めることでイベントのみを比較できます。階層内の最低レベルのイベントを検索するための条件を指定します。イベントの比較関数は、一致した各イベントと同じ行に親イベントの情報を含めることができます。たとえば、Phase1、P1、PhaseX、およびPhaseAのようなまったく異なる名前の第1フェーズイベントがあると仮定します。この場合、各フェーズイベントを比較して、親起動イベントの情報を含めることができます。

親イベント情報を含めることによるイベントの比較

イベント名	イベント テンプレート	期間	イベント名	イベント テンプレート	期間	RPM
TurbineStartUP.3.3	TurbineStartUp	01:42:00	Phase1	StartUpPhase1	00:35:00	1035
TurbineStartUP.4.3	TurbineStartUp	00:21:00	P1	StartUpPhase1	00:15:00	1075
TurbineStartUP.5.3	TurbineStartUp	01:55:00	Phase1	StartUpPhase1	00:31:00	1095
TurbineStartUP.6.3	TurbineStartUp	02:13:00	PhaseX	StartUpPhase1	00:23:00	1046
TurbineStartUP.8.1	TurbineStartUp	02:50:00	PhaseA	StartUpPhase1	00:28:00	850
TurbineStartUP.8.1	TurbineStartUp	02:50:00	PhaseA	StartUpPhase1	01:21:00	1334

エレメントに関連するイベントの検索

Explore Events 関数を使用すると、特定の PI AF エレメントに関連する分析を検索して分析できます。たとえば、特定のボイラーの停止時間を分析する必要がある場合、そのボイラーに関するすべての停止時間イベントを取得し、Microsoft Excel に貼り付けてグラフを作成することにより、データを分析できます。この場合、エレメント名とイベント名が一致するイベントを見つける必要があります。

1. イベントを含む関数配列の挿入開始セルをワークシート上で選択します。
2. [PI DataLink]タブの[イベント]グループで、[検索]をクリックして、[イベントの検索]タスクペインを開きます。
3. 該当するイベントを見つけるための条件を指定します。

条件を入力すると、[プレビュー]リストが更新され、指定した条件に一致するイベントが表示されます。

- a. [データベース]フィールドに、イベントを格納する PI AF データベースを入力します。

\ServerName\DatabaseName の形式で指定します。フィールドをクリックして、接続されている PI AF Server の、イベントテンプレートを含むデータベースのリストを開きます。

- b. [検索開始]フィールドと[検索終了]フィールドで、アクティブイベントを検索する期間を指定します。

PI 時間書式を入力します。たとえば、先月においてアクティブであったイベントを取得するには、「*-1mo」を[検索開始]フィールドに、「*」を[検索終了]フィールドに入力します。

ヒント: この期間に開始または終了したイベントなど、この期間に対してさらに特定の関連性を持つイベントを見つけるには、[その他の検索オプション]を展開し、[検索モード]リストから別の方を選択します。

- c. 必要に応じて条件を追加して、[プレビュー]リストに表示される、関数により返されるイベントを絞り込みます。

利用可能なフィールドの完全なリストについては、「[\[イベントの検索\]タスクペインリファレンス](#)」を参照してください。

たとえば、[イベントテンプレート]リストを使用して、返されるイベントを、特定のイベントテンプレートに基づくものに制限します。TurbineStartUp という名前のテンプレートを用意して、タービン起動イベントを分析するために選択できるようにすることができます。

同様に、[エレメント名]フィールドを使用して、返されるイベントを、特定のエレメントに関連するものに制限できます。「Boiler5」と入力することにより、このボイラーに関連するイベントを分析できます。

4. ワークシート出力を指定します。

a. [表示列]リストから、関数配列に含める列を選択し、列の順序を設定します。リストには、属性の名前が含まれます。既定では、すべてのイベントに対して生成される仮想属性と選択したイベントテンプレートからのイベント属性が、リストに含まれます。次の操作を実行できます。:

- リストされているすべての属性を、返される関数配列の列として含めるには、[すべて選択] チェックボックスを選択します。
- 特定の属性を含めるには、対応するチェックボックスを選択します。特定の属性を、返される関数配列内の列としては含めない場合は、対応するチェックボックスをオフにします。
-  をクリックして[属性の追加] ウィンドウを開きます。このウィンドウでは、返される関数配列内の列として追加する属性を選択できます。「[\[イベントの検索\]タスクペインへの属性列の追加](#)」を参照してください。
- リストの下部にある空白のチェックボックスの隣に、イベント属性の名前を入力します。
- 属性を右クリックし、[属性の追加]をクリックして、選択した属性の上に空白の属性を挿入します。次に、イベント属性の名前を入力します。
- 属性を選択し、 をクリックして、属性をリストの上位に移動します。
- 属性を選択し、 をクリックして、属性をリストの下位に移動します。
- 属性を選択し、 をクリックして、属性をリストから削除します。
- 属性を右クリックし、次に[属性の削除]をクリックして、属性をリストから削除します。

選択した列名の長さが 759 文字を超えると、関数は列を個別に処理できなくなります。この場合は、タスクペインに、すべての列をグループとして指定するよう求めるメッセージが表示されます。列をグループとして指定すると、リストには[すべてのイベント属性とテンプレート属性]と表示され、返される関数配列には、デフォルトの属性とその子の属性がすべて含まれますが、ユーザーが指定した追加の属性は含まれません。

b. [表示セル]フィールドに、関数配列の左上隅となるセルが設定されていることを確認します。

タスクペインを開く前にセルをクリックした場合、PI DataLink によりそのセルが自動的にこのフィールドに設定されます。

5. [OK]をクリックして、関数配列をワークシートに挿入します。

イベント名	開始時刻	終了時刻	プライマリ エレメント	ReasonCode	ShutDown
20140621 19:26	21-Jun-14 19:26:00	21-Jun-14 19:29:00	Boiler5	P	Planned
20140621 19:28	21-Jun-14 19:28:20	21-Jun-14 21:10:20	Boiler5	P	Planned
20140621 20:19	21-Jun-14 20:19:51	21-Jun-14 20:45:51	Boiler5	E	Emergency
20140621 20:26	21-Jun-14 20:26:00	21-Jun-14 20:29:00	Boiler5	P	Planned

Excel の機能を使用して、データを分析します。たとえば、グラフを作成できます。

[イベントの検索]タスクペインへの属性列の追加

[属性の追加] ウィンドウを使用して、[イベントの検索]タスクペインの [表示列] リストにイベント属性を追加します。これにより、ワークシートに挿入された関数配列にそれらの属性を含めることができます。

1. [イベントの検索]タスクペインを開き、ワークシートに取得するイベントを指定します。
2. タスクペインの [表示列] リストの隣にある  をクリックして、[属性の追加] ウィンドウを開きます。このウィンドウには、タスクペインで現在指定されている条件に一致するイベントが一覧表示されます。
3. イベントを拡張して、イベント属性がそのイベントのために格納されていることを確認します。関数配列に任意のイベント属性を列として含めることができます。関数配列に含めると、属性列にイベント用に格納された値が表示されます。
4. タスクペインの [表示列] リストに追加する属性の隣にあるチェックボックスを選択します。イベントの検索関数は名前で属性を識別します。イベントおよび属性名は、値を一意に定義します。そのため、特定の属性を追加するのは一度のみで済みます。属性を選択すると、イベント階層での位置に関係なく、その属性がウィンドウにより自動的に選択されます。

注: PI DataLink は、このウィンドウでの選択を、タスクペインの [表示列] リストと同期させません。このウィンドウでは、既に使用可能な、またはタスクペインの列リストで選択されている属性を選択できます。これにより、同じ属性をタスクペインの列リストに複数回追加できます。

5. [OK] をクリックして、[表示列] リストに属性を挿入し、その列を選択します。

子イベントのあるイベントの検索

Explore Events 関数を使用すると、子イベントのあるイベントを階層書式で分析できます。たとえば、タービンの起動フェーズを分析する必要があり、各フェーズは全体的な起動イベントの子イベントとして構成されている場合、すべてのタービン起動イベントを子イベント(フェーズイベント)とともに取得して Microsoft Excel に貼り付け、そのデータを分析できます。この場合、イベントテンプレートを使用してイベントを取得できます。

1. イベントを含む関数配列の挿入開始セルをワークシート上で選択します。
2. [PI DataLink] タブの [イベント] グループで、[検索] をクリックして、[イベントの検索] タスクペインを開きます。
3. 該当するイベントを見つけるための条件を指定します。

条件を入力すると、[プレビュー] リストが更新され、指定した条件に一致するイベントが表示されます。

- a. [データベース]フィールドに、イベントを格納する PI AF データベースを入力します。
\\ServerName\DatabaseName の形式で指定します。フィールドをクリックして、接続されている PI AF Server の、イベントテンプレートを含むデータベースのリストを開きます。
- b. [検索開始]フィールドと[検索終了]フィールドで、アクティブイベントを検索する期間を指定します。
PI 時間書式を入力します。たとえば、先月においてアクティブであったイベントを取得するには、「*-1mo」を[検索開始]フィールドに、「*」を[検索終了]フィールドに入力します。

ヒント : この期間に開始または終了したイベントなど、この期間に対してさらに特定の関連性を持つイベントを見つけるには、[その他の検索オプション]を展開し、[検索モード]リストから別の方法を選択します。

- c. 必要に応じて条件を追加して、[プレビュー]リストに表示される、関数により返されるイベントを絞り込みます。

利用可能なフィールドの完全なリストについては、「[\[イベントの検索\]タスクペインリファレンス](#)」を参照してください。

たとえば、[イベントテンプレート]リストを使用して、返されるイベントを、特定のイベントテンプレートに基づくものに制限します。TurbineStartUp という名前のテンプレートを用意して、タービン起動イベントを分析するために選択できるようにすることができます。

4. ワークシート出力を指定します。

- a. [表示列]リストから、関数配列に含める列を選択し、列の順序を設定します。リストには、属性の名前が含まれます。既定では、すべてのイベントに対して生成される仮想属性と選択したイベントテンプレートからのイベント属性が、リストに含まれます。次の操作を実行できます。:

- リストされているすべての属性を、返される関数配列の列として含めるには、[すべて選択] チェックボックスを選択します。
- 特定の属性を含めるには、対応するチェックボックスを選択します。特定の属性を、返される関数配列内の列としては含めない場合は、対応するチェックボックスをオフにします。
-  をクリックして[属性の追加]ウィンドウを開きます。このウィンドウでは、返される関数配列内の列として追加する属性を選択できます。「[\[イベントの検索\]タスクペインへの属性列の追加](#)」を参照してください。
- リストの下部にある空白のチェックボックスの隣に、イベント属性の名前を入力します。
- 属性を右クリックし、[属性の追加]をクリックして、選択した属性の上に空白の属性を挿入します。次に、イベント属性の名前を入力します。
 - 属性を選択し、 をクリックして、属性をリストの上位に移動します。
 - 属性を選択し、 をクリックして、属性をリストの下位に移動します。
 - 属性を選択し、 をクリックして、属性をリストから削除します。
 - 属性を右クリックし、次に[属性の削除]をクリックして、属性をリストから削除します。

選択した列名の長さが 759 文字を超えると、関数は列を個別に処理できなくなります。この場合は、タスクペインに、すべての列をグループとして指定するよう求めるメッセージが表

示されます。列をグループとして指定すると、リストには[すべてのイベント属性とテンプレート属性]と表示され、返される関数配列には、デフォルトの属性とその子の属性がすべて含まれますが、ユーザーが指定した追加の属性は含まれません。

b. [子イベントレベルの数]リストから、関数配列に含める子イベントレベルの数を選択します。

子イベントを取得しない場合は、[0]を選択します。第1レベルの子イベント（一致したイベント直下のイベント）を取得する場合は、[1]を選択します。以降、同様です。子イベントのあるイベントには、[プレビュー]リストで、イベント名の隣に+が表示されます。子イベントは、指定した条件に一致する必要はありません。

取得されるレベルごとに1つの列が追加されます。取得される子イベントは、それぞれ異なる行に挿入されます。子イベントを含む行には、子イベントの名前を含む列が追加されます。

c. [表示セル]フィールドに、関数配列の左上隅となるセルが設定されていることを確認します。

タスクペインを開く前にセルをクリックした場合、PI DataLinkによりそのセルが自動的にこのフィールドに設定されます。

5. [OK]をクリックして、関数配列をワークシートに挿入します。

イベント名	Child1	イベント テンプレート	開始時刻	終了時刻	プライマリ エレメン
TurbineStartup.1.1		TurbineStartUp	21-Jun-14 19:28:20	21-Jun-14 21:10:20	Turbine1
TurbineStartup.1.1	Phase1	StartUpPhase1	21-Jun-14 19:28:20	21-Jun-14 20:12:20	Turbine1
TurbineStartup.1.1	Phase2	StartUpPhase2	21-Jun-14 20:12:20	21-Jun-14 20:39:50	Turbine1
TurbineStartup.1.1	Phase3	StartUpPhase3	21-Jun-14 20:39:50	21-Jun-14 21:10:20	Turbine1
TurbineStartup.3.1		TurbineStartUp	21-Jun-14 21:10:20	21-Jun-14 22:25:20	Turbine1
TurbineStartup.3.1	Phase1	StartUpPhase1	21-Jun-14 21:10:20	21-Jun-14 21:15:20	Turbine1
TurbineStartup.3.1	Phase2	StartUpPhase2	21-Jun-14 21:15:20	21-Jun-14 21:22:20	Turbine1
TurbineStartup.3.1	Phase3	StartUpPhase3	21-Jun-14 21:22:20	21-Jun-14 22:25:20	Turbine1

子イベントを含めることによるイベントの比較

イベントの比較関数を使用すると、同じ名前の子イベントを持つイベントを比較できます。イベントの比較関数はパスを使用して属性を識別します。このため、この関数は、関連するイベントの属性を1つの行で返すことができます。たとえば、フェーズイベント（起動イベント時に発生する子イベント）を持つタービン起動イベントがあると仮定します。イベントの比較関数は、各起動イベントをそれぞれ1つの行で、対応するフェーズからの値を各行内の異なる列に表示できます。この返された関数配列を使用して、起動イベントを容易に比較できます。

1. イベントを含む関数配列の挿入開始セルをワークシート上で選択します。
2. [PI DataLink]タブの[イベント]グループで、[Compare (比較)]をクリックして、[イベントの比較]タスクペインを開きます。
3. 該当するイベントを見つけるための条件を指定します。

条件を入力すると、[プレビュー]リストが更新され、指定した条件に一致するイベントが表示されます。

a. [データベース]フィールドに、イベントを格納する PI AF データベースを入力します。

\ServerName\DatabaseName の形式で指定します。フィールドをクリックして、接続されている PI AF Server の、イベントテンプレートを含むデータベースのリストを開きます。

b. [検索開始] フィールドと [検索終了] フィールドで、アクティブイベントを検索する期間を指定します。

PI 時間書式を入力します。たとえば、先月においてアクティブであったイベントを取得するには、「*-1mo」を [検索開始] フィールドに、「*」を [検索終了] フィールドに入力します。

ヒント: この期間に開始または終了したイベントなど、この期間に対してさらに特定の関連性を持つイベントを見つけるには、[その他の検索オプション] を展開し、[検索モード] リストから別の方法を選択します。

c. 必要に応じて条件を追加して、[プレビュー] リストに表示される、関数により返されるイベントを絞り込みます。

利用可能なフィールドの完全なリストについては、「[Compare Events\] タスクペインリファレンス](#)」を参照してください。

たとえば、[イベントテンプレート] リストを使用して、返されるイベントを、特定のイベントテンプレートに基づくものに制限します。TurbineStartUp という名前のテンプレートを用意して、タービン起動イベントを分析するために選択できるようにすることができます。

4. [表示列] リストを使用して、返される関数配列の列を指定します。

リストには、属性の名前が含まれます。既定では、すべてのイベントに対して生成される仮想属性と選択したイベントテンプレートからのイベント属性が、リストに含まれます。関数はパスを使用して属性を識別します。サポートされているパス注釈の詳細については、「[イベントの比較関数のパス注釈](#)」を参照してください。

a.  をクリックして [属性の追加] ウィンドウを開き、親イベント間で比較する子イベントの属性を選択します。

[子イベント属性を列として\[イベントの比較\] タスクペインに追加](#) を参照してください。

b. 関数配列に含める必要のある親イベントの属性を挿入します。

[親イベント属性を列として\[イベントの比較\] タスクペインに追加](#) を参照してください。

c. 関数配列に含める列を選択し、列の順序を設定します。

次の操作を行うことができます。

- リストされているすべての属性を、返される関数配列の列として含めるには、[すべて選択] チェックボックスを選択します。
- 特定の属性を含めるには、対応するチェックボックスを選択します。特定の属性を、返される関数配列内の列としては含めない場合は、対応するチェックボックスをオフにします。
- リストの下部にある空白のチェックボックスの隣に、イベント属性の名前を入力します。
- 属性を右クリックし、[属性の追加] をクリックして、選択した属性の上に空白の属性を挿入します。次に、イベント属性の名前を入力します。

- 属性を選択し、 をクリックして、属性をリストの上位に移動します。
- 属性を選択し、 をクリックして、属性をリストの下位に移動します。
- 属性を選択し、 をクリックして、属性をリストから削除します。

- 属性を右クリックし、次に[属性の削除]をクリックして、属性をリストから削除します。

5. [表示セル]フィールドに、関数配列の左上隅となるセルが設定されていることを確認します。

タスクペインを開く前にセルをクリックした場合、PI DataLink によりそのセルが自動的にこのフィールドに設定されます。

6. [OK]をクリックして、関数配列をワークシートに挿入します。

イベント名	イベント テンプレート	開始時刻	終了時刻	期間	¥Phase1	¥Phase2
TurbineStartUp.3	TurbineStartUp	06-Jun-14 12:29:00	06-Jun-14 14:11:00	01:42:00	0 0:12:00	0 0:55:00
TurbineStartUp.5	TurbineStartUp	08-Jun-14 10:21:00	08-Jun-14 10:42:00	00:21:00	0 0:07:00	0 0:00:00

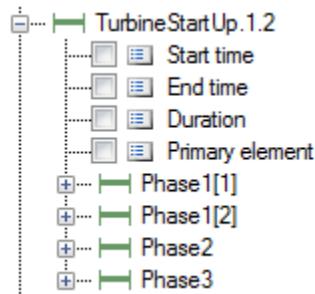
子イベント属性を列として[イベントの比較]タスクペインに追加

[属性の追加]ウィンドウを使用して、[イベントの比較]タスクペインの[表示列]リストに子イベント属性を追加します(これにより、ワークシートに挿入された関数配列にそれらの列を含めることができます)。タスクペインでイベントに条件を指定すると、条件に一致したイベント、およびその子イベントがウィンドウに一覧表示されます。このウィンドウを使用して、これらのイベントから任意のイベント属性を追加できます。通常、このウィンドウは一致したイベントを比較するのに役立つ子イベント属性を追加するために使用します。

- [イベントの比較]タスクペインを開き、ワークシートに取得するイベントを指定します。
- タスクペインの[表示列]リストの隣にある  をクリックして、[属性の追加]ウィンドウを開きます。このウィンドウには、タスクペインで現在指定されている条件に一致するイベントが一覧表示されます。
- イベントを拡張して、そのイベント用に格納されているイベント属性と、すべての子イベントを確認します。

同じ名前の子イベントが 2 つ以上ある場合、PI DataLink が子イベント名にインデックスを自動的に追加します。インデックスを作成するために、PI DataLink はこれらのイベントを開始時間の昇順、次に終了時間の昇順、その次に識別子の昇順で並び替えます。

Phase1 という名前の 2 つの子イベントがあるイベント

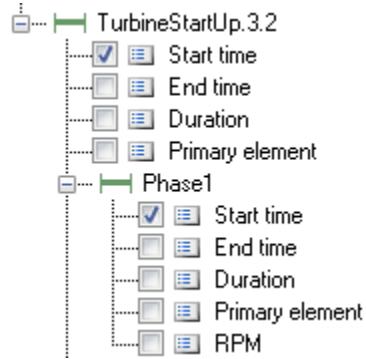


関数配列に任意のイベント属性を列として含めることができます。関数配列に含まれると、列にイベント属性用に格納された値が表示されます。

- タスクペインの[表示列]リストに追加する属性の隣にあるチェックボックスを選択します。

イベントの比較関数は名前、および一致したイベントに相対したパスで属性を識別します。たとえば、一致したイベントおよびその子イベントのそれぞれの **Start time** 属性を選択すると仮定します。

一致したイベントおよび子イベントの開始時刻属性を選択します。



これにより、2つの列が追加されます:

- .|Start time

ワークシートでは、この列には各行に一覧表示された一致したイベントの開始時刻が含まれています。

- .\Phase1|Start time

ワークシートでは、この列には各行に一覧表示された一致したイベントの Phase1 子イベントの開始時刻が含まれています。

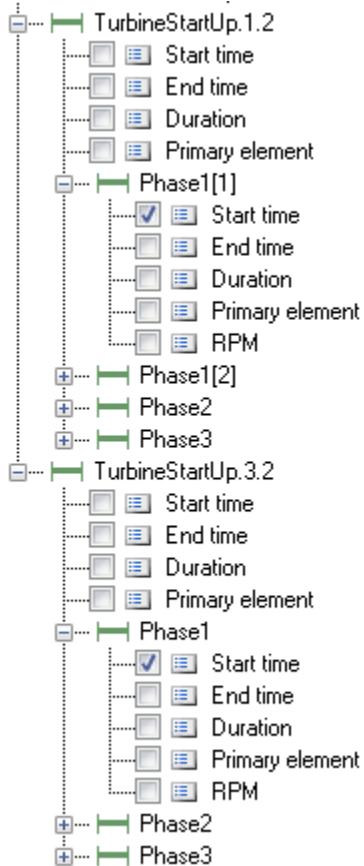
子属性列を含むワークシート

. Event name	. Start time	.\Phase1 Start time
TurbineStartUp.3.2	20-Dec-13 14:38:00	20-Dec-13 14:39:00
TurbineStartUp.9.2	12-Feb-14 11:07:00	12-Feb-14 11:07:00

階層内のどこからでも属性を追加できます。ウィンドウは選択を同期しません:パスと属性名が同じ場合、同じ属性を複数回追加できます。

イベントに一貫性のないイベント階層がある場合、子イベントの属性を選択する際は注意してください。同じ名前の複数の子イベントを含んでいるイベントがある場合、重複した子イベント(つまり、インデックス付き子イベント)の属性を選択して、すべてのイベントのデータが表示されるようにします。たとえば、重複した子イベント(TurbineStartUp.1.2 イベントの Phase1[1] 子イベント)および重複していない子イベント(TurbineStartUp.3.2 イベントの Phase1 子イベント)の配下の **Start time** 属性を選択すると仮定します。

重複した子イベント、および重複していない子イベントの開始時刻属性を選択します



これにより、異なる結果を返す 2 つの列が追加されます:

- .\Phase1[1]|Start time

ワークシートでは、この列には各行に一覧表示された、一致したイベントの最初の Phase1 子イベントの開始時刻が含まれています。一致したイベントにこの名前の子イベントのみがある場合、この列にそのイベントの開始時刻が一覧表示されます。

- .\Phase1|Start time

この列には各行に一覧表示された一致したイベントの唯一の Phase1 子イベントの開始時刻が表示されます。一致したイベントにこの名前の子イベントが複数ある場合、列に *Duplicate event exists: specify index for Phase1* が表示されます。

インデックス付き子属性とインデックスなし子属性を含むワークシート

. Event name	.\Phase1[1] Start time	.\Phase1 Start time
TurbineStartUp.5.2	20-Dec-13 07:15:00	Duplicate event exists: specify index for Phase1
TurbineStartUp.1.2	20-Dec-13 08:15:00	Duplicate event exists: specify index for Phase1
TurbineStartUp.3.2	20-Dec-13 14:39:00	20-Dec-13 14:39:00

インデックス付き属性は、必要なデータを返します。

5. [OK]をクリックして、[表示列]リストに属性を挿入し、その列を選択します。

親イベント属性を列として[イベントの比較]タスクペインに追加

ワークシートでイベントを比較しやすくするよう、一致したイベントの親イベントの情報を含む列を含めることができます。[イベントの比較]タスクペインの[表示列]リストを使用して、親イベント(およびそれらの親イベント、または階層内の任意の上位レベルイベント)の属性を挿入します。

注: イベントの比較関数は、イベントフレーム参照を介して定義された関係を持つイベントの親イベント属性からデータを取得しません。

1. [イベントの比較]タスクペインを開き、ワークシートに取得するイベントを指定します。
2. [イベントの比較]タスクペインの[表示列]リストで、属性に適した方法を使用して親イベント属性を追加します。

挿入するには、	次の操作を行います:
親レベルでの同じ属性名	<p>a. 属性を右クリックし、[親イベントのクローン]をクリックします。 PI DataLink は同じ名前の親イベント属性を選択した属性の上に挿入します。</p> <p>たとえば、[. \Event name]属性(一致したイベントの Event name 属性)を右クリックして、[親イベントのクローン]をクリックすると、PI DataLink は[.. \Event name]属性(親イベントの Event name 属性)を挿入します。</p> <p><input checked="" type="checkbox"/> .. \Event name <input checked="" type="checkbox"/> . \Event name</p> <p>操作を繰り返して、イベントの同じ属性を階層のより上位のイベントに追加できます。たとえば、[.. \Event name]属性を右クリックして、[親イベントのクローン]をクリックすると、PI DataLink は[..\\.. \Event name]属性(祖父イベントの Eventname 属性)を挿入します。</p> <p><input checked="" type="checkbox"/> ..\\.. \Event name <input checked="" type="checkbox"/> .. \Event name <input checked="" type="checkbox"/> . \Event name</p>
固有の属性名	<p>a. 親イベント属性を挿入する属性を右クリックして、[属性の挿入]をクリックします。</p> <p>PI DataLink は、プライマリイベントレベル(.)で空白の属性を挿入します。</p> <p>b. 親イベント属性の正しいパスと名前を入力します。</p> <p>たとえば、親イベントの Type 属性を含めるには、「.. Type」を入力します。</p> <p><input checked="" type="checkbox"/> .. \Event name <input checked="" type="checkbox"/> .. Type <input checked="" type="checkbox"/> . \Event name</p>

パス構文の詳細については、[イベントの比較関数のパス注釈](#)を参照してください。

親イベントを含めることによるイベントの比較

イベントの比較関数を使用することにより、あるイベントと同じ行に親イベントに関する情報を表示できます。この機能を使用すると、異なる名前を持つイベントを含めて、一致する階層からのあらゆるイベントを比較できます。たとえば、フェーズイベントを持つタービン起動イベントがあり、いくつかのフェーズイベントには異なる名前が割り当てられていると仮定します。イベントの比較関数を使用すると、各フェーズイベントをそれぞれ 1 つの行で、親イベントからの値を各フェーズイベント行内の異なる列に表示できます。返された情報を使用することにより、フェーズイベントを容易に比較できます。

1. イベントを含む関数配列の挿入開始セルをワークシート上で選択します。
2. [PI DataLink] タブの [イベント] グループで、[Compare (比較)] をクリックして、[イベントの比較] タスクペインを開きます。
3. 該当するイベントを見つけるための条件を指定します。

条件を入力すると、[プレビュー] リストが更新され、指定した条件に一致するイベントが表示されます。

- a. [データベース] フィールドに、イベントを格納する PI AF データベースを入力します。
\\ServerName\DatabaseName の形式で指定します。フィールドをクリックして、接続されている PI AF Server の、イベントテンプレートを含むデータベースのリストを開きます。
- b. [検索開始] フィールドと [検索終了] フィールドで、アクティブイベントを検索する期間を指定します。
PI 時間書式を入力します。たとえば、先月においてアクティブであったイベントを取得するには、「*-1mo」を [検索開始] フィールドに、「*」を [検索終了] フィールドに入力します。

ヒント : この期間に開始または終了したイベントなど、この期間に対してさらに特定の関連性を持つイベントを見つけるには、[その他の検索オプション] を展開し、[検索モード] リストから別 の方法を選択します。

- c. 必要に応じて条件を追加して、[プレビュー] リストに表示される、関数により返されるイベントを絞り込みます。

利用可能なフィールドの完全なリストについては、「[\[Compare Events\] タスクペインリファレンス](#)」を参照してください。

たとえば、[イベントテンプレート] リストを使用して、返されるイベントを、特定のイベントテンプレートに基づくものに制限します。StartUpPhase1 という名前のテンプレートを用意して、第 1 フェーズ起動イベントを分析するために選択できるようにすることができます。

4. [表示列] リストを使用して、返される関数配列の列を指定します。

リストには、属性の名前が含まれます。既定では、すべてのイベントに対して生成される仮想属性と選択したイベントテンプレートからのイベント属性が、リストに含まれます。関数はパスを使用して属性を識別します。サポートされているパス注釈の詳細については、「[イベントの比較関数のパス注釈](#)」を参照してください。

- a. 関数配列に含める必要のある親イベントの属性を挿入します。

[親イベント属性を列として\[イベントの比較\] タスクペインに追加](#) を参照してください。

たとえば、仮想属性のいくつかを右クリックし、[親イベントのクローン] をクリックすることにより、次のような、対応する親イベントの属性を挿入できます

..|Event name
..|Start time
..|Primary element

b. 関数配列に含める列を選択し、列の順序を設定します。

次の操作を行うことができます。

- リストされているすべての属性を、返される関数配列の列として含めるには、[すべて選択] チェックボックスを選択します。
- 特定の属性を含めるには、対応するチェックボックスを選択します。特定の属性を、返される関数配列内の列としては含めない場合は、対応するチェックボックスをオフにします。
- リストの下部にある空白のチェックボックスの隣に、イベント属性の名前を入力します。
- 属性を右クリックし、[属性の追加]をクリックして、選択した属性の上に空白の属性を挿入します。次に、イベント属性の名前を入力します。
- 属性を選択し、 をクリックして、属性をリストの上位に移動します。
- 属性を選択し、 をクリックして、属性をリストの下位に移動します。
- 属性を選択し、 をクリックして、属性をリストから削除します。
- 属性を右クリックし、次に[属性の削除]をクリックして、属性をリストから削除します。

5. [表示セル] フィールドに、関数配列の左上隅となるセルが設定されていることを確認します。

タスクペインを開く前にセルをクリックした場合、PI DataLink によりそのセルが自動的にこのフィールドに設定されます。

6. [OK]をクリックして、関数配列をワークシートに挿入します。

.. イベント名	.. イベント テンプレート	.. 期間	.. イベント名	.. イベント テンプレート	.. 期間	.. RPM	.. ラ
TurbineStartUP.3.3	TurbineStartUp	0 1:42:00	Phase1	StartUpPhase1	0 0:35:00	1035	Turb
TurbineStartUP.4.3	TurbineStartUp	0 0:21:00	P1	StartUpPhase1	0 0:15:00	1075	Turb
TurbineStartUP.5.3	TurbineStartUp	0 1:55:00	Phase1	StartUpPhase1	0 0:31:00	1095	Turb
TurbineStartUP.6.3	TurbineStartUp	0 2:13:00	PhaseX	StartUpPhase1	0 0:23:00	1046	Turb
TurbineStartUP.8.1	TurbineStartUp	0 2:50:00	PhaseA	StartUpPhase1	0 0:28:00	850	Turb
TurbineStartUP.8.1	TurbineStartUp	0 2:50:00	PhaseA	StartUpPhase1	0 1:21:00	1334	Turb

特定の親イベントのイベントを検索する

Explore Events 関数や Compare Events 関数を使用すると、特定の親イベントから派生したイベントに限定して情報を表示できます。特定の親イベントを基準にし、検索したイベントをしづら込むには、[関数タスク]ペインを開き、[データベース]フィールドで、親イベントへのパスを指定します。セル参照を使用してパスを指定できます。

1. 親イベントへのイベントパスをワークシートに取得します。

- Explore Events 関数を使用して親イベントを検索します。
- [表示列]リストで[イベントパス]チェックボックスのオン/オフで出力に含める列を指定します。

他の列を含めることもできます。

c. [OK] をクリックし、ワークシートにデータを取得します。

イベント名	イベントパス
TurbineStartUp.1.1	\\MyServer\MyDB\EventFrames[TurbineStartUp.1.1]
Phase1	\\MyServer\MyDB\EventFrames[TurbineStartUp.1.1]\Phase1
TurbineStartUp.3.1	\\MyServer\MyDB\EventFrames[TurbineStartUp.3.1]
Phase1	\\MyServer\MyDB\EventFrames[TurbineStartUp.3.1]\Phase1
Phase2	\\MyServer\MyDB\EventFrames[TurbineStartUp.1.1]\Phase2

2. Explore Events または Compare Events 関数で親イベントパスを参照します。

a. 同一または別のワークシートで、子イベントも含む、PI DataLink の関数配列の挿入開始セルをクリックします。
新しい配列が他の関数配列と重なってはならないことに注意してください。

b. [PI DataLink] タブの[イベント]グループで、[検証]または[Compare (比較)]をクリックし、対応する関数タスクペインを開きます。

c. [データベース]フィールドで、現在のデータベースへの既存のエントリをクリアし、[イベントパス]列で、イベント検索の基準にする親イベントに対応するセルをクリックします。

PI DataLink により、ワークシートのセルに対するセル参照がタスクペインに挿入されます。



3. 目的の子イベントと列を指定する条件がほかにあれば指定し、[OK]をクリックすると、ワークシートに関数配列が挿入されます。

予約属性名

PI DataLink は、イベント用に仮想属性を自動的に生成し、これらの仮想属性を識別するために予約名を使用します。イベント定義属性との競合を避けるために、属性名として以下の予約名を使用するイベントは定義しないでください。

- Acknowledged by
- Acknowledged date
- Annotations (アノテーション)
- Duration(期間)
- Element template
- End time
- Event categories

- Event name
- Event path
- Event template
- Is acknowledged
- Primary element
- Primary element path
- Severity
- Start time

章 5

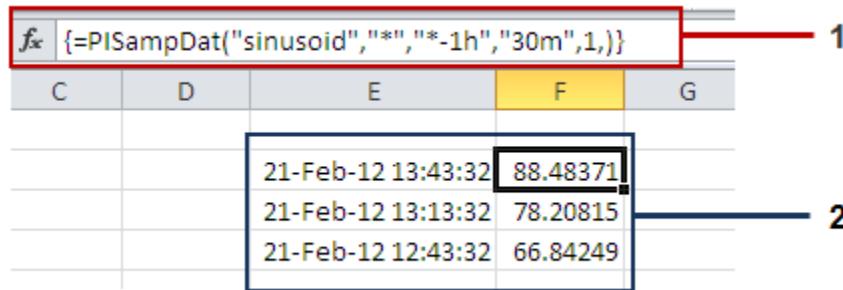
ワークシートの使用法とメンテナンス

本セクションでは、PI DataLink 関数がワークシート内でどのように機能するか、および作成したワークシートを管理/メンテナンスする方法について説明します。

関数配列

関数配列は、1つのPI DataLink 関数からの出力内容が含まれるセルのかたまりです。ワークシートに関数を追加すると、PI DataLink は対象の出力セルに関数式を入力します。関数は PI Data Archive Server や PI AF Server に問い合わせを行い、関数配列として値を取得します。配列の大きさは、関数、検索に使用する値の数、リクエストした値の数、リクエストした出力により異なります。

ワークシートは、返却値を関数配列のセルに表示します。ただし、実際にセルに入力されているのは関数式です。セルをクリックすると、Excel の数式バーに関数式が表示されます。



The screenshot shows an Excel spreadsheet with a DataArray formula in cell F1: `=PIISampDat("sinusoid","*","*-1h","30m",1,)`. The formula is highlighted with a red box and labeled '1'. Below it, a blue box highlights the returned data values in cells F2, F3, and F4, labeled '2'. The data consists of timestamped values: 21-Feb-12 13:43:32, 88.48371; 21-Feb-12 13:13:32, 78.20815; and 21-Feb-12 12:43:32, 66.84249.

- 1.数式バーの関数式
- 2.関数配列の返却値

共通関数配列タスク

関数配列は、ワークシート上で移動、コピー、削除できます。

配列を選択する方法は複数あります:

- 配列内のすべてのセルをドラッグして選択します。配列に空の値がある場合、それも選択する必要があります。
- 配列のセルをすべて選択するには、配列の一部を右クリックして、[DataLink 関数の選択]を選択します。

配列を選択すると、配列全体を容易に管理できます:

- 数式バーに表示される、頭記号\$を手動で削除すると、セルを相対参照できます。

注: 数式バーで変更後は、**Ctrl + Shift + Enter** キーを押して変更を適用します。

- 配列の任意のセルを右クリックし、**[切り取り]**、**[コピー]**、**[削除]**、**[数式と値のクリア]**、および**[セルの書式設定]**から希望の操作を選択します。
- コピーまたは切り取りを行った配列を再挿入するには、新しいセルを右クリックして**[貼り付け]**を選択します。
- 選択した配列の端にカーソルを置き、十字カーソルが表示されたら、配列を新しいセルにドラッグして移動します。

注: 関数配列は、すべてのセルを含む、配列全体を移動または削除する必要があります。そうしないと、PI DataLink は次のエラーメッセージを返します: **You cannot change part of an array.**

- 新しい位置に配列を配置するには、関数タスクペインを開き、**[表示セル]** フィールドを更新します。
[「関数配列の入力の更新」](#) を参照してください。

ワークシートの他の場所で関数配列の値を使用する場合、関数配列の値もコピーできます。コピーした配列を関数配列ではなく値として貼り付けた場合、関数配列の一部ではなく、単なるセルの値となるため、その値は更新されません。配列値のコピー方法は、以下のとおりです:

- 関数値を選択してコピーし、**[形式を選択して貼り付け]** コマンドを利用して、希望の場所に貼り付けます。

値をコピーする際、配列全体の行や列を選択する必要はありません。

関数配列の入力の更新

新しい入力値で関数配列を更新するには、関数タスクペインを使用します。

1. 配列の関数タスクペインを開きます。

- タスクペインの自動表示がオンになっている場合、1つのセルまたは複数のセルをクリックします。
- タスクペインの自動表示がオフになっている場合、1つのセルまたは複数のセルを右クリックし、関数名を選択します。

関数タスクペインに、選択した関数配列の現在の入力値が表示されます。

1つのセルを選択した場合、PI DataLink は隣接する関数配列からソース(データアイテムまたは式)のみが異なっているものを自動的に検知し、編集用にその配列を選択します。

複数のセルを選択した場合、PI DataLink は選択セルと同じソース(データアイテムまたは式の値が同じ)の配列を選択します。

2. 選択した配列の入力値を変更し、**[OK]** または **[適用]** をクリックします。

[表示セル] フィールドが変更された場合、結果は新しいセル参照の位置により異なります:

- 新しいセル参照が元の配列の一部である場合は、配列の左上端のセルを新しい参照セルに移動し、配列全体を移動します。
- 新しいセル参照が元の配列の一部でない場合は、PI DataLink は配列をコピーし、新しい参照セルが配列の左上端となるように貼り付けます。元の配列は、そのままの位置に保持されます。

配列の大きさ

関数配列を貼り付ける際、PI DataLink 関数は、返すデータに合わせて自動的に配列の大きさを変更します。関数が関数配列を貼り付けるのは、タスクペイン上で[OK]または[適用]をクリックしたとき、または[DataLink 関数の再計算 (サイズ変更)]をショートカットメニューでクリックしたときです。

関数の再計算と更新を行う他の方法では、新しい関数配列の貼り付けは行われません。このような方法では、配列値の更新のみを行い、配列のサイズ変更は行いません。たとえば、関数の入力値をセル参照で指定し、そのセルの値を変更した場合、関数は返却値を更新しますが、配列の大きさは変更されません。更新により、ワークシート上の関数配列で表示できるよりも多くの値(または少ない値)が返されることがあります。関数の返却値に関する設定:

- 更新によって、関数の表示可能数より少ない値が返された場合、PI DataLink は、値のないセルに対しブランクまたは#N/A のいずれかを表示します。
- 更新によって、関数の表示可能数より多くの値が返された場合、PI DataLink 関数は `Resize to show all values` というテキストを配列の下に表示します(参照設定でメッセージが無効になっている場合を除きます)。

空の値によって、ワークシートの外観が変わることがあります固定のサイズを維持するには、返却データの数を制限するように関数の入力値を設定してください。たとえば、圧縮データ関数では、期間を指定してその期間中の全データを取得するのではなく、取得する値の数を指定できます。

次の

[演算頻度](#)

[配列のサイズ変更](#)

[Excel の PI DataLink 設定を管理する](#)

配列のサイズ変更

PI DataLink 関数の更新や再計算では、関数配列が表示できる値数より多く取得されることもあるが、少なく取得されることもあります。[DataLink 関数の再計算 (サイズ変更)]コマンドを使用して、関数配列全体を更新します。DataLink は、PI Data Archive や PI AF から新しい値を取得し、取得したデータに合わせて配列のサイズを自動的に変更します。

関数配列のセルで右クリックして、[DataLink 関数の再計算 (サイズ変更)]をクリックします。

関数は現在の入力値を使って関数配列を再書き込みし、取得データに合わせて配列のサイズを自動的に調整します。

動的関数配列

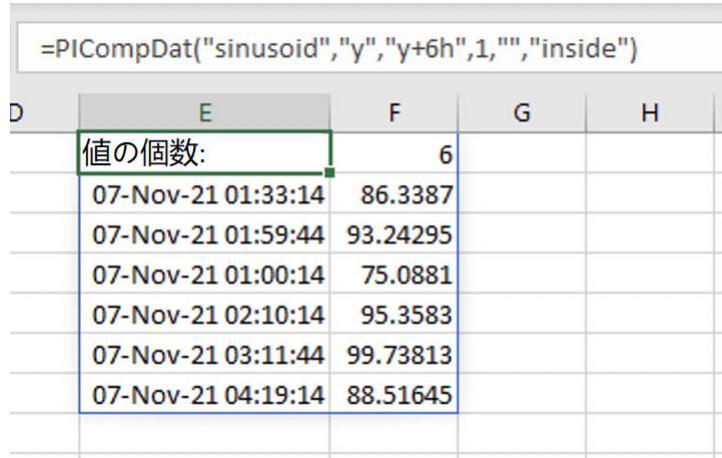
Office 2021 と Office 365 (2020 年 1 月 ビルド以降) より、Microsoft は動的配列のサポートを導入しました。動的配列に含まれるデータが更新されると、動的配列はサイズ変更されます。たとえば、圧縮されたデータ配列に前日に発生したイベントが表示され、そのイベント数が次回に増加した場合、ユーザーの操作なしでスプレッドシートが開いて動的配列がサイズ変更されます。

DataLink はデフォルト設定ではレガシ関数配列を書き込み続けます。しかし動的配列がサポートされていると、ユーザーはレガシ関数配列で右クリックして動的配列に変換するオプションを使用できます。この変換を行った後、それに続いて行われる関数へのタスクペインを介したいかなる変更も、シートに動的配列として書き込まれます。

動的配列を含むワークブックを、動的配列に対応していない古いバージョンの Excel で開くと、動的配列はレガシ関数配列として利用可能になります。そのため、ワークブックを使用する全ユーザーがこの機能をサポートするエクセルバージョンを所有している場合のみ動的配列を使用することを強くお勧めします。

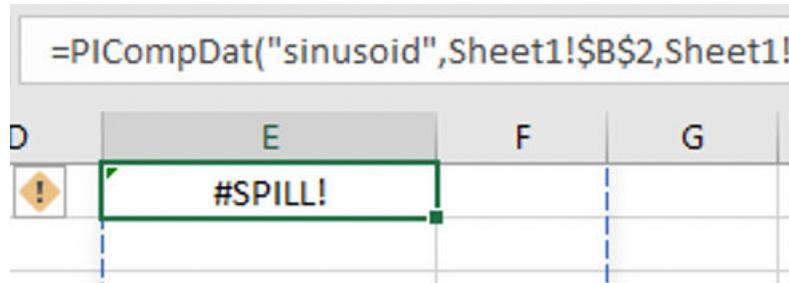
動的配列の挙動

動的配列は、そのコンテンツが配列の **SpillParent** と呼ばれる動的配列の左上セルの式によってのみ決定される点がレガシ関数配列と異なります。関数出力を保持する配列の残り部分は、**SpillingToRange** と呼ばれます。下図で、緑線で囲まれたセルが **SpillParent** で、青線で囲まれた全領域が **SpillingToRange** です。 **SpillingToRange** 内の他のセルをクリックすると、数式バーの式を変更できないことが分かります。



The screenshot shows an Excel spreadsheet with a formula bar containing the formula =PICompDat("sinusoid","y","y+6h",1,"","inside"). The table has columns D, E, F, G, and H. Row D contains the formula. Row E contains the header "値の個数:" and the value 6. Rows F through H contain data: F1: 07-Nov-21 01:33:14, F2: 86.3387; F3: 07-Nov-21 01:59:44, F4: 93.24295; F5: 07-Nov-21 01:00:14, F6: 75.0881; F7: 07-Nov-21 02:10:14, F8: 95.3583; F9: 07-Nov-21 03:11:44, F10: 99.73813; F11: 07-Nov-21 04:19:14, F12: 88.51645. The cells in row E and the entire data range from F1 to F12 are highlighted with a green border, indicating they are part of the spill range. The cells in row D are highlighted with a blue border, indicating they are the spill parent.

スプレッドシートの領域内に動的配列が適合しない場合は、左上の **SpillParent** セルのみに#SPILL! というメッセージが表示されます（下のスクリーンショット参照）。



The screenshot shows an Excel spreadsheet with a formula bar containing the formula =PICompDat("sinusoid",Sheet1!\$B\$2,Sheet1!. The table has columns D, E, F, and G. The first cell (D1) contains a yellow warning icon. The cell E1 contains the error message "#SPILL!". The entire row E1 is highlighted with a green border, indicating it is the spill range. The cell D1 is highlighted with a blue border, indicating it is the spill parent.

動的配列に変換する

PI DataLink はデフォルト設定ではレガシ関数配列を書き込みます。関数配列を動的配列に変換するには、右クリックの**[動的配列に変換]**コマンドを使用します。

関数配列のセルを右クリックして、**[動的配列に変換]**をクリックします。関数が関数配列を動的配列に書き換えます。それに続く関数への更新も、動的配列としてシートに書き込まれます。



演算頻度

関数タスクペインで[OK]または[Apply]をクリックすると、PI DataLink は関数配列を書き込み、関数の現在値を計算します。データを最新の状態に維持するには、関数配列を再計算する必要があります。以下のセクションでは、関数配列を再計算する方法を説明します。

Excel 計算に関する詳細は、MSDN ライブラリの [Excel 再計算](#) を参照してください。

自動更新機能

自動更新機能は、一定の間隔で関数とトレンドの再計算を自動的に行います。自動更新は、同じ Excel セッションで開かれたすべてのワークブックに適用されます。メニュー、タスクペイン、またはウィンドウを開いたとき、または、現在のワークシートが編集モードにある場合は、自動更新は一時停止されます。自動更新機能は、読み取り専用である保護ブックの更新は行いません。

自動更新は設定により影響されます:

- 更新タイプ

自動更新時にどの関数を更新するかを決定する設定。2つの選択肢があります:

- 計算(F9 キー)

この設定を選択すると、再計算時にすべての揮発性関数と、揮発性関数を参照するすべての関数が再計算されます。

- 全計算(Ctrl+Alt+Shift+F9)

この設定を選択すると、再計算時に、揮発性に関係なく、すべての関数を再計算します。

- 更新間隔

[間隔]フィールドで、自動計算の間隔を秒数で入力します。最小値は 5 秒です。0 を入力すると、PI DataLink は自動的に計算間隔を 2 倍に設定します(最小間隔は 5 秒)。自動間隔では、Excel が計算に要する時間の 2 倍分の時間が確保されるため、計算間隔の半分の時間は、Excel がユーザーの作業(データ入力等)を処理できる状態を確保できます。

設定した間隔より再計算の時間の方が長くなつた場合、自動間隔に設定を切り替えるか、自動更新を無効にすることを促すメッセージが表示されます。

次の

[Excel の PI DataLink 設定を管理する](#)

[自動更新の有効化](#)

[自動更新の無効化](#)

自動更新の有効化

自動更新機能を有効にすると、指定した間隔で PI DataLink 関数が自動的に再計算されます。

[PI DataLink]タブで、[更新]ボタンをクリックします。

Excel は、現在の Excel セッションで開いているすべてのワークブックにある PI DataLink 関数を即座にすべて再計算します。その後、Excel は、指定した間隔で自動的に関数の再計算を行います。

自動更新機能を有効にすると、[更新]ボタンは強調表示され、ステータスバーにメッセージが表示されます。

次の

[Excel の PI DataLink 設定を管理する](#)

自動更新の無効化

自動更新機能を無効にして、PI DataLink 関数の自動再計算を停止します。

1. [PI DataLink]タブで、強調表示されている[更新]ボタンをクリックし、選択を解除します。
2. 再計算処理中にキャンセルする場合は、**Esc** キーを押します。PI DataLink は現在処理中の関数の計算を終了し、未処理の関数配列セルには計算中止と表示して、自動更新を無効にします。

トリガーされた再計算

ワークシートのセルを編集時、または **F9** を押すと、Excel は、揮発性関数と、揮発性関数を参照する関数またはチャートが再計算されます。揮発性関数には、Excel 時間関数の **now()** と **today()** が含まれます。現在値関数は揮発性関数です。他の PI DataLink 関数は揮発性関数ではありません。非揮発性の PI DataLink 関数内で揮発性関数を参照することで、非揮発性の PI DataLink 関数の再計算を実行できます。

揮発性 Excel 関数を含む PI DataLink 関数

PI DataLink 関数内で揮発性 Excel 関数を参照することによって、更新頻度を最大にして、関数配列とチャートを最新状態で維持できます。

注: 自動更新機能を使用していない場合に、更新頻度を最大にするには、ワークシートとアプリケーションの両方を [Manual] 計算ではなく [Automatic] 計算に設定してください (Excel の計算方法と設定を確認してください)。ただし、パフォーマンスに影響するため、ワークシートが大きい場合は操作が困難になる場合があります。

1. Excel の数式バーから、ワークシートのセルに揮発性関数を入力します。

たとえば、(**today()**+1/3) で今日の午前 8 時を表すことや、**now()**をセル参照として使用し、現在の PI 時間 (*) と置き換えることができます。

更新を開始する絶対タイムスタンプを作成するには、2-feb-12 00:30:30 + **now()** - **now()** のように **now()** の加算と減算を行います。

2. DataLink 関数の[Start time]入力や[End time]入力を指定する際に、そのセルを参照します。

手動による再計算

PI DataLink 関数は、いつでも手動で再計算できます。

- F9 キーを押すと、すべての揮発性関数(と揮発性関数を参照する関数)は強制的に再計算されます。
- すべての PI DataLink 関数を強制的に再計算するには、Ctrl + Alt + Shift + F9 キーを押します。
- 関数配列の任意の場所を右クリックし、次に**[DataLink 関数の再計算 (サイズ変更)]**をクリックします。このコマンドは、関数配列を書き込み、返されたデータに合わせて関数配列のサイズを自動的に変更します。

ワークシートの共有

PI System データを他のユーザーと共有するには、PI DataLink で作成したワークシートを共有します。ワークシートを他のユーザーと共有する場合には、次のことを考慮してください:

- 他の Microsoft Excel ユーザーに PI DataLink ワークシートを送信できます。受信したユーザーが関数を再計算し、PI ポイント値や PI AF 属性値を動的に更新して確認するには、PI DataLink をインストールし、同じ PI Data Archive Server と PI AF Server に接続する必要があります。
- PI DataLink がインストールされていない場合でも、ワークシートに最後に保存されたデータを閲覧できます。ただし、Excel でワークシートを開く前に、[ブックの計算]オプションが[手動]に設定されていなければなりません。「[\[ブックの計算\]オプションの\[手動\]設定](#)」を参照してください。
- ワークシートを、Web ページや PDF ファイルとして保存することで、静的ワークシートを共有できます。
- **[形式を選択して貼り付け]**コマンドを使用することで、関数配列の値をコピーして、ワークシートの新しい場所に貼り付けることができます。ワークシートのデータは、コピーされると再計算できなくなりますが、この方法によって、Excel を利用できるユーザーならだれにでも、カスタマイズされたデータスナップショットを配布できます。「[共通関数配列タスク](#)」を参照してください。

[ブックの計算]オプションの[手動]設定

PI DataLink をインストールしていないユーザーが、ワークブックに保存された最後のデータを確認できるようにするには、ワークブックを保存する前に、Excel の[ブックの計算]オプションを[手動]に設定します。

1. [ファイル]タブをクリックし、次に[オプション]を選択します
2. [Excel のオプション]ウィンドウで[数式]をクリックします。
3. [計算方法の設定]で、[ブックの計算]を[手動]に設定して、[OK]をクリックします。

章 6

PI DataLink 関数

PI DataLink 関数を使用することで、PI Data Archive や PI AF Server を検索し、取得した検索結果に計算を適用して、ワークシートに値を返すことができます。他の Excel 関数のように、PI DataLink 関数の結果は関数配列で返されます。このため、必要に応じて再計算し値を更新できます。関数タスクペインで PI DataLink 関数の入力値を定義し、関数をワークシートに貼り付けます。

本セクションでは、各関数の入力値について説明します。

次の

[関数タスクペイン](#)

[関数配列](#)

[関数リファレンス](#)

現在値関数

現在値関数は、履歴データ用 PI タグや PI AF 属性の現在値や最新値を取得します。未来の PI ポイントの場合、戻り値は、現在の時刻を基準に記録されている値のタイムスタンプによって決定します。

- 記録されている値がすべて、現在の時刻より前の場合、この関数で返される値は、最後に記録された値です。
- 記録されている値のタイムスタンプが、現在の時刻をまたいでいる場合、この関数は、内挿値を返します。
- 記録されている値がすべて、現在の時刻より後の場合、この関数でデータは返されません。

現在値関数は揮発性関数です。Excel がワークシート内のどこかのセルを計算、再計算すると、現在値関数も値を再計算し更新します。すぐに再計算を実行するには、F9 キーを押します。

この関数は一括呼び出しをサポートしています。

入力	説明
ルートパス	指定データアイテムへの共有パス。有効な項目は以下のものを含みます: <ul style="list-style-type: none"> PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。 PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。 ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 「 データアイテム 」を参照してください。
データアイテム	関数が値を返す PI ポイント名または PI AF 属性名。 複数のデータアイテムを指定するには、PI ポイント名や PI AF の属性名を含むセル範囲を入力します。この関数では、それぞれに値が返されます。必須。
表示セル	関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。 関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。 複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。
タイムスタンプなし	値だけを返す場合は、このオプションをクリックします。参照データアイテムが列の場合、関数は値を列で返します。参照データアイテムが行の場合、関数は値を行で返します。
タイムスタンプ左	このオプションをクリックすると、返される値は 2 列表示になります。左の列にはタイムスタンプ、右の列には値が表示されます。
タイムスタンプ上	このオプションをクリックすると、返される値は 2 行表示になります。上の行にはタイムスタンプ、下の行には値が表示されます。

次の

[PICurrVal\(\)](#)

[大規模データの取得](#)

[内挿値](#)

現在値関数の例

PI ポイント `sinusoid` の現在値を取得するには、現在値関数に次の引数を設定します。

入力	値
データアイテム	<code>sinusoid</code>

入力	値
タイムスタンプ左	選択

関数は次の配列を返します。

28-Aug-12 10:40:38	18.07261276
--------------------	-------------

ヒストリ値関数

- 【データアイテム】オプションを選択した場合、圧縮データ関数は、指定したタイムスタンプにおける PI ポイントや PI AF 属性の値を返します。この関数は一括呼び出しをサポートしています。
- 【式】オプションを選択した場合、ヒストリ値関数は、指定したタイムスタンプにおける式の計算値を取得します。

入力	説明
ルートパス	指定したデータアイテムへの共有パス(式内のデータアイテムを含む)。有効な項目は以下のものを含みます: <ul style="list-style-type: none"> PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。 PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。 ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 「 データアイテム 」を参照してください。
データアイテム	関数が値を返す PI ポイント名または PI AF 属性名。 複数のデータアイテムを指定するには、PI ポイント名や PI AF の属性名を含むセル範囲を入力します。この関数では、それぞれに値が返されます。 【データアイテム】オプションを選択した場合は必須。
式	関数が値を計算するパフォーマンス イクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。 【式】オプションを選択した場合は必須。
タイムスタンプ	タイムスタンプ。関数は、これに対する値を返します。固定時刻、または現在時刻に対する時間式のいずれかを指定します。「 時間入力 」を参照してください。 必須。

入力	説明
取得モード	<p>返却値を決定するために関数が使用する方法。以下の選択肢があります。</p> <ul style="list-style-type: none"> 前の値 指定した時刻と正確に一致する値、またはそれ以前の値を返します。 前の値のみ 指定した時刻以前の値を返します。 内挿値 指定した時刻の内挿値を取得します。 「内挿値」を参照してください。 自動 指定した時刻の内挿値を取得します。または、[前の値]と同じ結果になります。(ステップ属性を持ったデータアイテムに対しては[前の値]を使用します。) 次の値 指定した時刻と正確に一致する値、またはそれ以前の値を返します。 次の値のみ 指定した時刻以降の値を返します。 指定時刻 指定した時刻と正確に一致する値を返します。指定時刻の値が存在しない場合は、<code>No events found</code> が返されます。 <p>[データアイテム]オプションを選択した場合に使用可能。</p>
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>
タイムスタンプなし	値だけを返す場合は、このオプションをクリックします。参照データアイテムまたは式が列の場合、関数は値を列で表示します。参照データアイテムまたは式が行の場合、関数は値を行で表示します。
タイムスタンプ左	このオプションをクリックすると、返される値は 2 列表示になります。左の列にはタイムスタンプ、右の列には値が表示されます。
タイムスタンプ上	このオプションをクリックすると、返される値は 2 行表示になります。上の行にはタイムスタンプ、下の行には値が表示されます。

次の

- [PIArcVal\(\)](#)

データアイテムを指定します。

- [PIExpVal\(\)](#)

式を指定します。

- [大規模データの取得](#)

ヒストリ値関数の例

ある時刻の PI ポイント `sinusoid` の値を取得するには、ヒストリ値関数に次の引数を設定します。

入力	値
データアイテム	<code>sinusoid</code>
タイムスタンプ	<code>30-Sep-07 15:13</code>
取得モード	<code>previous</code>
タイムスタンプ左	選択

関数は次の配列を返します。

30-Sep-07 14:39:16	99.18375
--------------------	----------

圧縮データ関数

- **[時間範囲]**オプションを選択した場合、圧縮データ関数は、指定した時間範囲にあるすべての PI ポイント値または PI AF 属性値、および値の総数を取得します。
- **[値の個数]**オプションを選択した場合、圧縮データ関数は、指定時刻で始まるポイント値や属性値を指定した数だけ取得します。

注: ヒストリデータは、圧縮アルゴリズムによって同じ傾斜を表す値が削除された後に、PI Data Archive のアーカイブに格納されたデータです。PI Data Archive のスナップショットに格納された値は、この圧縮アルゴリズムを経ていません。圧縮データ関数の時間範囲内に現在時刻を指定した場合、関数はその時間に対するスナップショット値を取得します。したがって、取得したスナップショット値は、圧縮によりアーカイブに格納されたものではありません。

入力	説明
ルートパス	<p>指定したデータアイテムへの共有パス(式内のデータアイテムを含む)。有効な項目は以下のものを含みます:</p> <ul style="list-style-type: none">• PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。• PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。• ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 <p>「データアイテム」を参照してください。</p>
データアイテム	<p>関数が値を返す PI ポイント名または PI AF 属性名。</p> <p>複数のデータアイテムを指定するには、PI ポイント名や PI AF の属性名を含むセル範囲を入力します。この関数では、それぞれに値が返されます。必須。</p>
開始時刻	<p>関数が値を返す期間の開始時刻。「時間入力」を参照してください。必須。</p>
終了時刻	<p>関数が値を返す期間の終了時刻。「時間入力」を参照してください。</p> <p>[時間範囲]オプションを選択した場合は必須。</p>
値の個数	<p>関数が取得する値の数。[開始時刻]で始まります。</p> <p>[値の個数]オプションを選択した場合は必須。</p>
後方検索	<p>このチェックボックスを選択すると、[開始時刻]以前に記録された値が返されます。[値の個数]オプションを選択し、[値の個数]フィールドに値を入力すると使用可能になります。</p> <p>[値の個数]フィールドにセル参照を入力すると、チェックボックスは無効になります。セル参照に負の値を入力することで、同じ結果が得られます。</p>
フィルター式	<p>関数が値をフィルターするときに使用するブール型のパフォーマンスイクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「フィルター式」を参照してください。</p>
フィルター済みの表示	<p>このチェックボックスを選択すると、ラベル <code>Filtered</code> が、[フィルター式]により結果から除外された値の位置に表示されます。</p>

入力	説明
境界タイプ	<p>開始時刻/終了時刻付近のどの値を取得するか決定するために関数が使用する方法:</p> <ul style="list-style-type: none">• 内側(既定) 開始時刻と終了時刻に値が存在する場合、その値を取得します。値が存在しなければ、期間内で発生した直近の値を取得します。• 外側 期間外で発生した直近の値を取得します。• 内挿値 開始時刻と終了時刻の内挿値を取得します。 「内挿値」を参照してください。• 自動 内挿値を返します。ただし、ステップ属性のデータアイテムの場合、<code>inside</code> が使用されます。
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>
値の個数 非表示	このチェックボックスを選択すると、取得された値の数を非表示にし、取得された値のみを表示します。[時間範囲]オプションを選択した場合に使用可能。
タイムスタンプの表示	各返却値に対してタイムスタンプを表示するには、このチェックボックスを選択します。1列に値を表示していた場合は、その左側にタイムスタンプを表示します。1行に値を表示していた場合は、その上側にタイムスタンプを表示します。

入力	説明
値属性の表示	<p>取得された値に関連する拡張ステータスビットを表示するには、このチェックボックスを選択します。ステータスビットには以下のようなものがあります。</p> <ul style="list-style-type: none"> • A Annotated = イベントにコメントが追加されていることを示します。 • S Substituted = 値が記録された元の値から変更されていることを示します。 • Q Questionable = 何らかの理由で値の精度に問題があることを示します。 <p>PI DataLink は、値属性を値の右側の列に表示します。</p>
注釈の表示	取得された値に関連する注釈を表示するには、このチェックボックスを選択します。注釈には、データ値を説明するノートやコメントが含まれます。PI DataLink は、注釈を値の右側の列に表示します。
列	1列に値を返したい場合は、このオプションをクリックします。
行	1行に値を返したい場合は、このオプションをクリックします。 注: [データアイテム]に複数項目 (一定のセル範囲) を指定した場合、または[式]に複数式 (一定のセル範囲) を指定した場合、PI DataLink は、指定範囲の方向に合わせて自動的に[列]または[行]を選択します。

次の

- [PICompDat\(\)](#)
フィルター式を使用せずに時間範囲を指定します。
- [PICompFilDat\(\)](#)
フィルター式を使用して時間範囲を指定します。
- [PINCompDat\(\)](#)
フィルター式を使用せずに複数の値を指定します。
- [PINCompFilDat\(\)](#)
フィルター式を使用して複数の値を指定します。

圧縮データ関数の例

PI ポイント `sinusoid` の直近 10 個の値を取得するには、圧縮データ関数に次の引数を設定します。

入力	値
値の個数	選択
データアイテム	sinusoid
開始時刻	\$A\$10 (セル A10 には現在時刻が入力されていると仮定)
値の個数	10
後方検索	選択
境界タイプ	inside
タイムスタンプの表示	選択
列	選択

関数は次の配列を返します。

27-Aug-12 14:19:38	96.92956
27-Aug-12 13:33:08	86.30853
27-Aug-12 10:40:38	18.07261
27-Aug-12 09:31:38	1.893003
27-Aug-12 08:31:38	1.524215
27-Aug-12 07:31:38	14.14445
27-Aug-12 04:58:08	75.70212
27-Aug-12 03:45:08	96.17171
27-Aug-12 02:40:08	99.25044
27-Aug-12 01:31:08	85.70325

[時間範囲]オプションを選択し、適切な[開始時刻]と[終了時刻]を指定することで、同じ値を取得することもできます。

定間隔圧縮データ関数

- [データアイテム]オプションを選択した場合、定間隔圧縮データ関数は、一定期間の PI ポイント値または PI AF 属性値を定間隔で内挿した値を返します。
- [式]オプションを選択した場合、定間隔圧縮データ関数は、一定期間のパフォーマンスイクエージョンの値を定間隔で内挿した値を返します。

[「内挿値」](#)を参照してください。

入力	説明
ルートパス	<p>指定したデータアイテムへの共有パス(式内のデータアイテムを含む)。有効な項目は以下のものを含みます:</p> <ul style="list-style-type: none">• PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。• PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。• ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 <p>「データアイテム」を参照してください。</p>
データアイテム	<p>関数が値を返す PI ポイント名または PI AF 属性名。</p> <p>複数のデータアイテムを指定するには、PI ポイント名や PI AF の属性名を含むセル範囲を入力します。この関数では、それぞれに値が返されます。</p> <p>[データアイテム]オプションを選択した場合は必須。</p>
式	<p>関数が値を計算するパフォーマンス イクエージョン。「式」を参照してください。</p> <p>複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。</p> <p>[式]オプションを選択した場合は必須。</p>
開始時刻	<p>関数が値を返す期間の開始時刻。「時間入力」を参照してください。</p> <p>必須。</p>
終了時刻	<p>関数が値を返す期間の終了時刻。「時間入力」を参照してください。</p> <p>必須。</p>
時間間隔	<p>関数が一定期間中に値を算出して返す頻度。値と時間単位を入力します。たとえば、15m (15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「時間間隔の仕様」を参照してください。</p> <p>必須。</p>
フィルター式	<p>関数が値をフィルターするときに使用するブール型のパフォーマンス イクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「フィルター式」を参照してください。</p>
フィルター済みの表示	<p>このチェックボックスを選択すると、ラベル <code>Filtered</code> が、[フィルター式] により結果から除外された値の位置に表示されます。</p>

入力	説明
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを【表示セル】に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を【表示セル】に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>
タイムスタンプの表示	各返却値に対してタイムスタンプを表示するには、このチェックボックスを選択します。1列に値を表示していた場合は、その左側にタイムスタンプを表示します。1行に値を表示していた場合は、その上側にタイムスタンプを表示します。
列	1列に値を返したい場合は、このオプションをクリックします。
行	<p>1行に値を返したい場合は、このオプションをクリックします。</p> <p>注: 【データアイテム】に複数項目（一定のセル範囲）を指定した場合、または【式】に複数式（一定のセル範囲）を指定した場合、PI DataLink は、指定範囲の方向に合わせて自動的に【列】または【行】を選択します。</p>

次の

- [PI SampDat\(\)](#)
フィルター式を使用してデータアイテムを指定します。
- [PI SampFilDat\(\)](#)
フィルター式を使用せずにデータアイテムを指定します。
- [PI ExpDat\(\)](#)
式を指定します。

定間隔ヒストリデータ関数の例

過去 24 時間の PI ポイント `sinusoid` に対する定間隔での内挿値を取得するには、定間隔ヒストリデータ関数に次の引数を指定します。

入力	値
データアイテム	<code>sinusoid</code>
開始時刻	<code>-1d</code>
終了時刻	<code>\$A\$10</code> (セル A10 には現在時刻が入力されていると仮定)
時間間隔	<code>3h</code>

入力	値
タイムスタンプの表示	選択
列	選択

関数は次の配列を返します。

26-Aug-12 14:03:42	92.40146637
26-Aug-12 17:03:42	73.10555267
26-Aug-12 20:03:42	7.732470989
26-Aug-12 23:03:42	26.54699326
27-Aug-12 02:03:42	92.09828186
27-Aug-12 05:03:42	73.467659
27-Aug-12 08:03:42	7.398549557
27-Aug-12 11:03:42	27.19914818
27-Aug-12 14:03:42	93.86413574

3 時間間隔の値が配列に表示されます。

時間指定圧縮データ関数

- [データアイテム]オプションを選択した場合、時間指定圧縮データ関数は、指定したタイムスタンプにおける PI ポイントや PI AF 属性の実際の値または内挿されたサンプル値を取得します。
- [式]オプションを選択した場合、時間指定圧縮データ関数は、指定したタイムスタンプにおける式の計算結果を返します。

入力	説明
ルートパス	<p>指定したデータアイテムへの共有パス(式内のデータアイテムを含む)。有効な項目は以下のものを含みます:</p> <ul style="list-style-type: none"> • PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。 • PI AF Server とデータベース (データ アイテムが PI AF 属性の場合)。 • ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 <p>「データアイテム」を参照してください。</p>
データアイテム	<p>関数が値を返す PI ポイント名または PI AF 属性名。</p> <p>複数のデータアイテムを指定するには、PI ポイント名や PI AF の属性名を含むセル範囲を入力します。この関数では、それぞれに値が返されます。</p> <p>[データアイテム]オプションを選択した場合は必須。</p>

入力	説明
式	関数が値を計算するパフォーマンス イクエージョン。「式」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。 [式]オプションを選択した場合は必須。
タイムスタンプ	関数が値を取得するタイムスタンプ。タイムスタンプを入力するか、タイムスタンプ値を含む 1 つ以上のセルを参照します。「 時間入力 」を参照してください。 必須。
取得モード	関数による値の取得方法: <ul style="list-style-type: none"> 内挿値 指定したタイムスタンプに対応する内挿値を返します。ステップ属性を持つデータアイテムの場合、指定したタイムスタンプより前の値を取得します。 「内挿値」を参照してください。 指定時刻 指定した時刻と正確に一致する値を取得します。指定時刻の値が存在しない場合は、メッセージ <code>No events found</code> が返されます。 [データアイテム]オプションを選択した場合に使用可能。
表示セル	関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。 関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。 複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。

次の

- [PITimeDat\(\)](#)

データアイテムを指定します。

- [PITimeExpDat\(\)](#)

式を指定します。

時間指定ヒストリデータ関数の例

タイムスタンプの配列に対応する PI ポイント `sinusoid` の値を取得するには、時間指定ヒストリデータ関数に次の引数を指定します。

入力	値
データアイテム	sinusoidu
タイムスタンプ	\$A\$3..\$A\$12 (セル A3 から A12 には、時間指定ヒストリデータ配列に対するタイムスタンプが入力されています)
取得モード	interpolated

関数は、以下の時間指定ヒストリデータを右側の列に取得します。

27-Aug-12 14:19:38	96.92956	0.806815326
27-Aug-12 13:33:08	86.30853	2.124000072
27-Aug-12 10:40:38	18.07261	58.63540268
27-Aug-12 09:31:38	1.893003	83.12264252
27-Aug-12 08:31:38	1.524215	96.88283539
27-Aug-12 07:31:38	14.14445	97.71135712
27-Aug-12 04:58:08	75.70212	48.53567886
27-Aug-12 03:45:08	96.17171	19.80453873
27-Aug-12 02:40:08	99.25044	3.524959087
27-Aug-12 01:31:08	85.70325	2.132091284

時間指定ヒストリデータ配列は、第 1 列にあるタイムスタンプに基づきます。

演算データ関数

- [データアイテム]オプションを選択した場合、演算データ関数は、PI ポイント値や PI AF 属性値と指定の演算設定を基に演算した定間隔の値を返します。フィルター式と時間間隔を指定しない場合、関数は一括呼び出しをサポートします。
- [式]オプションを選択した場合、演算データ関数は、パフォーマンスイクエージョンと指定の演算設定に基づいて演算した定間隔の値を返します。

注: [詳細設定]の隣にある「+」をクリックすると、すべての関数の入力値にアクセスできます。

入力	説明
ルートパス	<p>指定したデータアイテムへの共有パス(式内のデータアイテムを含む)。有効な項目は以下のものを含みます:</p> <ul style="list-style-type: none">• PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。• PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。• ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 <p>「データアイテム」 を参照してください。</p>
データアイテム	<p>関数が値を返す PI ポイント名または PI AF 属性名。</p> <p>複数のデータアイテムを指定するには、PI ポイント名や PI AF の属性名を含むセル範囲を入力します。この関数では、それぞれに値が返されます。</p> <p>[データアイテム]オプションを選択した場合は必須。</p>
式	<p>関数が値を計算するパフォーマンス イクエージョン。「式」を参照してください。</p> <p>複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。</p> <p>[式]オプションを選択した場合は必須。</p>
開始時刻	<p>関数が値を返す期間の開始時刻。「時間入力」を参照してください。</p> <p>必須。</p>
終了時刻	<p>関数が値を返す期間の終了時刻。「時間入力」を参照してください。</p> <p>必須。</p>
時間間隔	<p>関数が一定期間中に値を算出して返す頻度。値と時間単位を入力します。たとえば、15m (15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「時間間隔の仕様」 を参照してください。</p>
フィルター式	<p>関数が値をフィルターするときに使用するブール型のパフォーマンス イクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「フィルター式」 を参照してください。</p>

入力	説明
コンバージョンファクター	<p>関数が取得された値に適用する係数。コンバージョンファクターが不要の場合は、1を入力します。時間加重の積算値を計算する場合、記録周期を PI Server の既定時間単位(単位/日)に変換する係数を指定します。</p> <p>必須。</p> <p>たとえば、一般的なコンバージョンファクターには次のものがあります。</p> <ul style="list-style-type: none">• Units/day 1.0• Units/hour 24• Units/minute 1440• Units/second 86400
演算モード	<p>関数が実行する計算の種類。</p> <ul style="list-style-type: none">• 積算 期間内の値の合計。• 最小 期間内の最小値を見つけます。• 最大 期間内の最大値を見つけます。• 標準偏差 期間内の値の標準偏差を計算します。• 範囲 期間内の最大値から最小値を引いたものを計算します。• カウント [計算基準]フィールドが[時間加重]に設定されている場合は、期間内の秒数をカウントします。 [計算基準]フィールドが[イベント加重]に設定されている場合は、期間内に格納された値の数をカウントします。• 平均(時間加重) 期間内の記録値の時間加重平均を計算します。• 平均(イベント加重) 期間内の記録値のイベント加重平均を計算します。

入力	説明
計算基準	<p>計算方法:</p> <ul style="list-style-type: none"> 時間加重 イベント加重平均以外のすべての計算モードの既定値です。記録された各値に対して、その値が適用される時間の長さによって、重み付けが行われます。期間の境界にある値が内挿されます。 イベント加重 記録された各値は、等しく加重されます。バッチ値の場合、このオプションを選択します。この方法では、期間内に少なくとも 1 つの記録値(標準偏差の計算時には 2 つの記録値)が必要になります。 <p>[式]フィールドに式を入力し[式のサンプリング]フィールドを[内挿値]に設定すると、値が記録されない期間は関数によって値が補間されます。</p>
式のサンプリング	<p>式を評価するタイミングを決定するために関数が使用する方法:</p> <ul style="list-style-type: none"> 圧縮データ 関数が期間内の式の評価を行うのは、式中の PI ポイントや PI AF 属性が値を格納しているタイムスタンプにおいてです。 保存された値がない場合、関数では値が返されないか、補間値が返されます。期間内に保存された値が少ない場合に最適な結果を得るには、[圧縮済み]ではなく[内挿値]を選択します。 内挿値 関数は、[式のサンプリング間隔]で指定された一定のサンプリング間隔で式を評価します。 「内挿値」を参照してください。 <p>注: 結果の正確さを最大限に高めるには、[内挿値]を選択し、[式のサンプリング間隔]フィールドを低く設定します。</p> <p>[PI 式]を選択した場合、または[フィルター式]を指定した場合に使用可能です。</p>
式のサンプリング間隔	関数が式を評価する頻度。 [式のサンプリング] を [内挿値] に設定した場合には必須です。 たとえば、時間間隔を 10m (10 分) に設定すると、10 分ごとに内挿値が計算されます。
有効データ割合の最小値	各期間で値を計算し取得するために必要とされる有効データ割合の最小値。この値を満たさない期間に対しては、ラベル Insufficient good data が返されます。

入力	説明
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを【表示セル】に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を【表示セル】に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>
開始時刻の表示	このチェックボックスを選択すると、取得値の計算に使用する期間の開始時刻を表示します。開始時刻は、取得値の左の列または上の行に表示されます。【時間間隔】を指定したときのみ利用可能です。
終了時刻の表示	このチェックボックスを選択すると、取得値の計算に使用する期間の終了時刻を表示します。終了時刻は、取得値の左の列または上の行に表示されます。【時間間隔】を指定したときのみ利用可能です。
最小/最大時刻の表示	<p>このチェックボックスを選択すると、計算対象期間の最大値、最小値に対応するタイムスタンプが表示されます。【演算モード】の次の 3 つの値に対してのみ有効です:</p> <ul style="list-style-type: none">• 最大 最大値のタイムスタンプが表示されます。• 最小 最小値のタイムスタンプが表示されます。• 範囲 最大値と最小値のタイムスタンプを表示されます。
有効データ割合の表示	<p>このチェックボックスを選択すると、対象期間中に有効データが返された時間をパーセンテージで表示します。PI DataLink は、有効データ割合を取得値の右側の列か、下側の行に表示します。</p> <p>有効データとは、PI Data Archive により有効であると判断された、エラー状態ではない記録値です。有効データ割合は、PI ポイント値から構成される計算の信頼性を評価するために使用します(特に、計算した結果を他の計算で使用する場合)。</p> <p>たとえば、時間範囲に不正データが含まれている場合、時間加重による積算値は、積算値をデータが有効であった時間の割合で割ったものになります。こうした標準化は、データが不正であった期間の平均値が、期間全体の平均値と等しいことを前提としています。したがって、データの不正な期間が大きい場合には、平均値の信頼性は低くなります。</p>
列	1 列に値を返したい場合は、このオプションをクリックします。

入力	説明
行	1行に値を返したい場合は、このオプションをクリックします。 注: [データアイテム]に複数項目 (一定のセル範囲) を指定した場合、または[式]に複数式 (一定のセル範囲) を指定した場合、PI DataLink は、指定範囲の方向に合わせて自動的に[列]または[行]を選択します。

次の

- [PIAdvCalcVal\(\)](#)

データアイテムを指定して、値を 1 つ取得します。

- [PIAdvCalcFilVal\(\)](#)

データアイテムとフィルター式を指定して、値を 1 つ取得します。

- [PIAdvCalcExpVal\(\)](#)

式を指定して、値を 1 つ取得します。

- [PIAdvCalcExpFilVal\(\)](#)

式とフィルター式を指定して、値を 1 つ取得します。

- [PIAdvCalcDat\(\)](#)

データアイテムと時間間隔を指定して、複数の値を取得します。

- [PIAdvCalcFilDat\(\)](#)

データアイテム、時間間隔、およびフィルター式を指定して、複数の値を取得します。

- [PIAdvCalcExpDat\(\)](#)

式と時間間隔を指定して、複数の値を取得します。

- [PIAdvCalcExpFilDat\(\)](#)

式、時間間隔、およびフィルター式を指定して、複数の値を取得します。

- [大規模データの取得](#)

演算データ関数の例

今日の午前 0 時から現在時刻までの、1 時間ごとの PI ポイント `sinusoid` の値の範囲を取得するには、演算データ関数に次の引数を指定します。

入力	値
データアイテム	<code>sinusoid</code>
開始時刻	<code>t</code>
終了時刻	<code>*</code>
時間間隔	<code>1h</code>

入力	値
演算モード	range
計算基準	time-weighted
開始時刻の表示	選択
終了時刻の表示	選択
列	選択

関数は次の配列を返します。

27-Aug-12 00:00:00	27-Aug-12 01:00:00	24.07529
27-Aug-12 01:00:00	27-Aug-12 02:00:00	18.15997
27-Aug-12 02:00:00	27-Aug-12 03:00:00	7.87962
27-Aug-12 03:00:00	27-Aug-12 04:00:00	6.306436
27-Aug-12 04:00:00	27-Aug-12 05:00:00	17.04949
27-Aug-12 05:00:00	27-Aug-12 06:00:00	24.06163
27-Aug-12 06:00:00	27-Aug-12 07:00:00	24.06163
27-Aug-12 07:00:00	27-Aug-12 08:00:00	18.65239
27-Aug-12 08:00:00	27-Aug-12 09:00:00	6.653669
27-Aug-12 09:00:00	27-Aug-12 10:00:00	6.846051

この例では、関数は 1 時間間隔で"範囲"を計算します。どの演算にも、任意の時間間隔を指定できます。

条件指定時間データ関数

条件指定時間データ関数は、指定期間内でパフォーマンスイクエージョンが `true` と評価された合計時間を返します。

- 条件指定時間データ関数の結果は、PI Data Archive のバージョンにより、若干異なります。
- `TimeGE` や `TimeGT` といった `performance-equation` 関数や `asset-analytics` 関数の結果は、条件指定時間データ関数の結果より正確です。

条件指定時間データ関数の結果を使用して意思決定を行う前に、条件指定時間データ関数の結果を `performance-equation` 関数や `asset-analytics` 関数からの出力と比較することをお勧めします。

入力	説明
ルートパス	<p>指定したデータアイテムへの共有パス(式内のデータアイテムを含む)。有効な項目は以下のものを含みます:</p> <ul style="list-style-type: none">• PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。• PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。• ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 <p>「データアイテム」を参照してください。</p>
式	<p>関数が評価するブール型のパフォーマンス イクエージョンです。「式」を参照してください。</p> <p>複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。</p> <p>必須。</p>
開始時刻	<p>関数が値を返す期間の開始時刻。「時間入力」を参照してください。</p> <p>必須。</p>
終了時刻	<p>関数が値を返す期間の終了時刻。「時間入力」を参照してください。</p> <p>必須。</p>
時間間隔	<p>関数が一定期間中に値を算出して返す頻度。値と時間単位を入力します。たとえば、15m (15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「時間間隔の仕様」を参照してください。</p>
時間単位	<p>関数が値を取得する際に使用する時間単位。</p> <p>必須。</p>
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>
開始時刻の表示	<p>このチェックボックスを選択すると、取得値の計算に使用する期間の開始時刻を表示します。開始時刻は、取得値の左の列または上の行に表示されます。[時間間隔]を指定したときのみ利用可能です。</p>
終了時刻の表示	<p>このチェックボックスを選択すると、取得値の計算に使用する期間の終了時刻を表示します。終了時刻は、取得値の左の列または上の行に表示されます。[時間間隔]を指定したときのみ利用可能です。</p>

入力	説明
有効データ割合の表示	<p>このチェックボックスを選択すると、対象期間中に有効データが返された時間をパーセンテージで表示します。PI DataLink は、有効データ割合を取得値の右側の列か、下側の行に表示します。</p> <p>有効データとは、PI Data Archive により有効であると判断された、エラー状態ではない記録値です。有効データ割合は、PI ポイント値から構成される計算の信頼性を評価するために使用します(特に、計算した結果を他の計算で使用する場合)。</p> <p>たとえば、時間範囲に不正データが含まれている場合、時間加重による積算値は、積算値をデータが有効であった時間の割合で割ったものになります。こうした標準化は、データが不正であった期間の平均値が、期間全体の平均値と等しいことを前提としています。したがって、データの不正な期間が大きい場合には、平均値の信頼性は低くなります。</p>
列	1列に値を返したい場合は、このオプションをクリックします。
行	<p>1行に値を返したい場合は、このオプションをクリックします。</p> <p>注: 項目を含むセル範囲を[PI式]に指定すると、PI DataLink は、項目の方向に合わせて自動的に[列]または[行]を選択します。</p>

次の

- [PITimeFilterVal\(\)](#)

単一の値を返します。

- [PITimeFilter\(\)](#)

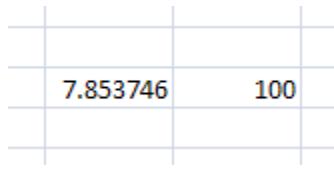
時間間隔を指定し、複数の値を返します。

条件指定時間データ関数の例

過去 7 日間で PI ポイント `sinusoid` が 75 を超えた時間数を確認するには、条件指定時間データ関数に次の引数を指定します。

入力	値
PI式	<code>'sinusoid' > 75</code>
開始時刻	<code>-7d</code>
終了時刻	<code>*</code>
時間単位	<code>h</code>
有効データ割合の表示	選択
列	選択

関数は次の配列を返します。



この例における計算結果では、75 を超える `sinusoid` の値は、過去 7 日間で 7.85 時間であり、この期間中の有効データ率は 100% となります。

イベントの検索関数

イベントの検索関数は、指定された条件に一致したイベントを PI AF データベースから取得して返します。階層書式でイベントを表示し調査するには、イベントの検索関数を使用します。この関数は行ごとに 1 つのイベントを返します。

[イベントの検索] タスクペインの一番上にあるフィールドで、取得する PI AF イベントを指定します。イベントに対して条件を指定すると、タスクペインの [プレビュー] リストに一致したイベントが表示されます。ワークシートに関数配列を挿入する前に、検索条件を修正して一致するイベントを変更できます。タスクペインの下にあるフィールドで、含める列、および挿入する関数配列の位置を指定します。

詳細については、以下のトピックを参照してください:

- [ワークシートのイベント](#)

PI DataLink は、イベントの検索関数、またはイベントの比較関数のいずれかを使用して、PI AF イベントから取得したデータを表示できます。

- [エレメントに関連するイベントの検索](#)

この手順に従って、イベントの検索関数を使用して、特定の PI AF エレメントに関連するイベントを分析します。

- [子イベントのあるイベントの検索](#)

この手順に従って、イベントの検索関数を使用して、子イベントを持つイベントを分析します。

- [\[イベントの検索\] タスクペインリファレンス](#)

タスクペインのフィールドでは、取得するイベント、およびワークシートに返す列とデータを指定できます。

- [予約属性名](#)

自動生成される属性との競合を避けるには、予約属性名を使用するイベントを定義しないようにします。

- [イベントの検索の例](#)

イベントの検索関数の入力値を設定して、過去 1 か月間アクティブで、特定のテンプレートに基づいたイベントを取得します。

- [Excel の PI DataLink 設定を管理する](#)

[設定] ウィンドウを使用して、関数が返すイベントの最大数を設定します。

[イベントの検索]タスクペインリファレンス

Explore Events 関数はイベントを階層書式で返します。設定を行うことにより、[プレビュー]リストに取得されるイベントの数とワークシートに返されるイベントの数を制限できます。

注: [その他の検索オプション]を展開すると、関数のすべての入力にアクセスできます。

入力	説明
[データベース]	イベントの取得先となる PI AF データベースです。`\\ \\ServerName\\DatabaseName` の書式で指定します。フィールドをクリックして、接続されている PI AF Server の、イベントテンプレートを含むデータベースのリストを表示します。データベースを指定しないと、関数は一致イベントを見つけることができません。 注: 特定の親イベント下に返されるイベントを制限するときは、[データベース]フィールドにその親イベントのイベントパスを指定します。「 特定の親イベントのイベントを検索する 」を参照してください。
[検索開始]	データベースでイベントの検索が始まる時刻を指定する PI 時間書式です。たとえば、「*-12h」と指定すると、12 時間前以降のデータベースレコードからイベントが検索されます。
[検索終了]	データベースでイベントの検索が終了する時刻を指定する PI 時間書式です。たとえば、「*」と指定すると、現時点までのイベントが検索されます。
[データベースレベルに限定]	データベースのルートレベルで一致イベントを検索する場合のみ、このチェックボックスを選択します。このチェックボックスをオフにすると、階層内のすべてのレベルで一致イベントが検索されます。
[イベント名]	一致対象のイベントの名前です。ワイルドカード文字を使用して、名前の一部を指定できます。
[イベントテンプレート]	一致対象のイベントのイベントテンプレートです。ベースイベントテンプレートを選択すると、派生したテンプレートからのイベントが含まれることに注意してください。 イベントテンプレートを選択すると、[表示列]リストが更新されて、選択したテンプレートの属性が表示されます。これにより、以前に挿入された属性はすべて削除されます。
[エレメント名]	一致対象のイベントにより参照される PI AF エレメントです。ワイルドカード文字を使用して、名前の一部を指定できます。PI AF Server バージョン 2.8 以降を使用している場合は、特定のエレメントのパスを指定できます。
[エレメントテンプレート]	一致対象のイベントにより参照されるエレメントのエレメントテンプレートです。ベースエレメントテンプレートを選択すると、派生したテンプレートからのエレメントを参照するイベントが含まれることに注意してください。 PI AF Server バージョン 2.6 以降では、エレメントテンプレートでフィルタ一処理する必要があります。

入力	説明
[イベントカテゴリ]	一致対象のイベントのカテゴリです。
[最小期間]	一致対象のイベントの最小期間です。値と時間単位の略号を指定します。
[最大期間]	一致対象のイベントの最大期間です。値と時間単位の略号を指定します。
[検索モード]	<p>検索開始と検索終了により指定された期間に相対して一致イベントを検索するために使用される方法です。</p> <ul style="list-style-type: none"> • active in range 指定した期間内の任意の時間にアクティブであったイベントを見つけています。 • entirely in range 指定した期間内に開始および終了したイベントを見つけています。 • starting in range 指定した期間内に開始し、期間内または期間後に終了したイベントを見つけています。 • ending in range 指定した期間内に終了し、期間前または期間内に開始したイベントを見つけています。 • in progress 指定した期間内に開始し、期間内に終了しなかったイベントを見つけています。PI AF Server バージョン 2.6 以降でのみ利用できます。
[並べ替え順序]	<p>返却値を並び替えるために関数が使用する方法。</p> <ul style="list-style-type: none"> • name ascending イベントをイベント名の昇順(A から Z)で並べます。 • name descending イベントをイベント名の降順(Z から A)で並べます。 • start time ascending イベントを開始時刻の古いものから新しいものの順に並べます。 • start time descending イベントを開始時刻の新しいものから古いものの順に並べます。 • end time ascending イベントを終了時刻の古いものから新しいものの順に並べます。 • end time descending イベントを終了時刻の新しいものから古いものの順に並べます。

入力	説明
[重大度]	一致対象のイベントの重要度です。演算子と重要度のタイプを指定します。指定された条件に基づいて一致対象イベントがフィルタリングされます。重要度のタイプは値に対応しており、重要度の高い方から順に表示されます。たとえば、「< Warning」と入力した場合、重要度が Warning 未満 (Information または None) のイベントが検索されます。
[属性値フィルター]	一致対象のイベントをフィルター処理するために使用できる、最大 4 つの属性条件です。 <u>注:</u> 属性値フィルターを指定する前に、イベントテンプレートを指定する必要があります。 各フィルターに対して、次を指定します。 <ul style="list-style-type: none">属性 一致対象のイベントをフィルター処理するために使用されるイベント属性です。利用できる属性は、選択したイベントテンプレートにより異なります。演算子 指定した属性値に適用される関係演算子です。利用できる演算子は、属性のデータタイプにより異なります。値 一致する属性を検索するために使用される値です。たとえば、[演算子] フィールドを [=] に設定した場合、イベントは、指定した属性がこの値に等しいものに制限されます。
[アックフィルター]	一致対象イベントをアック可能なイベントに制限します。 <ul style="list-style-type: none">アック済みのイベントを検索するには、[アック済] チェックボックスをオンにします。アックされていないイベントを検索するには、[未アック] チェックボックスをオンにします。現在のアック状態に関係なく、アック可能なすべてのイベントを検索するには、両方のチェックボックスをオンにします。
[注釈付きに制限]	注釈付きのイベントのみを検索するには、このチェックボックスをオンにします。注釈付きかどうかに関係なく、すべてのイベントを検索するには、このチェックボックスをオフにします。

入力	説明
[表示列]	<p>返される関数配列内の列。リストには、属性の名前が含まれます。既定では、すべてのイベントに対して生成される仮想属性と選択したイベントテンプレートからのイベント属性が、リストに含まれます。</p> <p>次の操作を行うことができます。</p> <ul style="list-style-type: none">リストされているすべての属性を、返される関数配列の列として含めるには、[すべて選択]チェックボックスを選択します。特定の属性を含めるには、対応するチェックボックスを選択します。特定の属性を、返される関数配列内の列としては含めない場合は、対応するチェックボックスをオフにします。 <p> をクリックして[属性の追加]ウィンドウを開きます。このウィンドウでは、返される関数配列内の列として追加する属性を選択できます。[イベントの検索]タスクペインへの属性列の追加を参照してください。</p> <ul style="list-style-type: none">リストの下部にある空白のチェックボックスの隣に、イベント属性の名前を入力します。属性を右クリックし、[属性の追加]をクリックして、選択した属性の上に空白の属性を挿入します。次に、イベント属性の名前を入力します。 <p> 属性を選択し、 をクリックして、属性をリストの上位に移動します。</p> <p> 属性を選択し、 をクリックして、属性をリストの下位に移動します。</p> <p>属性を選択し、 をクリックして、属性をリストから削除します。</p> <p>属性を右クリックし、次に[属性の削除]をクリックして、属性をリストから削除します。</p> <p>選択した列名の長さが 759 文字を超えると、関数は列を個別に処理できなくなります。この場合は、タスクペインに、すべての列をグループとして指定するよう求めるメッセージが表示されます。列をグループとして指定すると、リストには[すべてのイベント属性とテンプレート属性]と表示され、返される関数配列には、デフォルトの属性とその子の属性がすべて含まれますが、ユーザーが指定した追加の属性は含まれません。</p>
[子イベントレベルの数]	関数配列に含める子イベントレベルの数です。子イベントは、指定した条件に一致する必要はありません。子イベントはそれぞれ異なる行に返され、返されるレベルごとに列が追加されます。子イベントを含む行には、子イベントの名前を含む列が追加されます。

入力	説明
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>

次の

[ワークシートのイベント](#)

[\[イベントの検索\]タスクペインへの属性列の追加](#)

[Excel の PI DataLink 設定を管理する](#)

イベントの検索の例

イベントテンプレート PowerPlantShutDown に基づくイベントを PI AF Server AFSRV1 上のデータベース Production から取得するには、イベントの検索関数に次の引数を設定します:

入力	値
データベース	\\AFSRV1\\Production
検索 開始	*-1mo
検索 終了	*
イベント テンプレート	PowerPlantDownTime

関数は次の配列を返します。

Event name	Start time	End time	Duration	Event template	Prim
PowerPlantShutDown - 20130403.2	03-Apr-13 18:00:00	03-Apr-13 19:00:00	0 1:00:00	PowerPlantShutDown	Big C
PowerPlantShutDown - 20130404.2	04-Apr-13 18:00:00	04-Apr-13 19:00:00	0 1:00:00	PowerPlantShutDown	Big C
PowerPlantShutDown - 20130405.2	05-Apr-13 18:00:00	05-Apr-13 19:00:00	0 1:00:00	PowerPlantShutDown	Big C

イベントの比較関数

イベントの比較関数は、データベースで指定した条件に一致したイベントを返します。フラット書式でイベントを表示するには、イベントの比較関数を使用します。この関数は、行ごとに 1 つのイベントを返しますが、同じ行に関連するイベントの属性を返すこともできます。特に、イベントの比較を容易にするために、イベントの比較関数は返したイベントと同じ行に子イベントや親イベントから属性を返すことができます。

[イベントの比較]タスクペインの一番上にあるフィールドで、取得する PI AF イベントを指定します。イベントに対して検索条件を指定すると、タスクペインの[プレビュー]リストに一致したイベントが表示されます。ワークシートに関数配列を挿入する前に、検索条件を修正して一致するイベントを変更できます。タスクペインの下にあるフィールドで、含める列、および挿入する関数配列の位置を指定します。

詳細については、以下のトピックを参照してください:

- [ワークシートのイベント](#)

PI DataLink は、イベントの検索関数、またはイベントの比較関数のいずれかを使用して、PI AF イベントから取得したデータを表示できます。

- [子イベントを含めることによるイベントの比較](#)

この手順に従って、イベントの比較関数を使用して、同じ名前の子イベントを持つイベントを比較します。

- [親イベントを含めることによるイベントの比較](#)

この手順に従って、イベントの比較関数を使用して、子イベントが異なる名前を持つ場合に、一致する階層でイベントを比較します。

- [\[Compare Events\]タスクペインリファレンス](#)

タスクペインのフィールドでは、取得するイベント、およびワークシートに返す列とデータを指定できます。

- [イベントの比較関数のパス注釈](#)

属性名の特定のパス注釈を使用して、イベント階層の位置を指定します。

- [予約属性名](#)

自動生成される属性との競合を避けるには、予約属性名を使用するイベントを定義しないようにします。

- [イベントの比較の例](#)

イベントの比較関数の引数を設定して、温度警告イベントを比較します。

- [Excel の PI DataLink 設定を管理する](#)

[設定]ウィンドウを使用して、関数が返すイベントの最大数を設定します。

[Compare Events]タスクペインリファレンス

Compare Events 関数はイベントをフラット書式で返します。設定を行うことにより、[プレビュー]リストに取得されるイベントの数とワークシートに返されるイベントの数を制限できます。

注: [その他の検索オプション]の隣にある「+」をクリックすると、関数のすべての入力にアクセスできます。

入力	説明
[データベース]	<p>イベントの取得先となる PI AF データベースです。`\\ \\ServerName\\DatabaseName の書式で指定します。フィールドをクリックして、接続されている PI AF Server の、イベントテンプレートを含むデータベースのリストを表示します。データベースを指定しないと、関数は一致イベントを見つけることができません。</p> <p><u>注: 特定の親イベント下に返されるイベントを制限するときは、[データベース] フィールドにその親イベントのイベントパスを指定します。「特定の親イベントのイベントを検索する」を参照してください。</u></p>
[検索開始]	データベースでイベントの検索が始まる時刻を指定する PI 時間書式です。たとえば、「*-12h」と指定すると、12 時間前以降のデータベースレコードからイベントが検索されます。
[検索終了]	データベースでイベントの検索が終了する時刻を指定する PI 時間書式です。たとえば、「*」と指定すると、現時点までのイベントが検索されます。
[データベースレベルに限定]	データベースのルートレベルで一致イベントを検索する場合のみ、このチェックボックスを選択します。このチェックボックスをオフにすると、階層内のすべてのレベルで一致イベントが検索されます。
[イベント名]	一致対象のイベントの名前です。ワイルドカード文字を使用して、名前の一部を指定できます。
[イベントテンプレート]	<p>一致対象のイベントのイベントテンプレートです。ベースイベントテンプレートを選択すると、派生したテンプレートからのイベントが含まれることに注意してください。</p> <p>イベントテンプレートを選択すると、[表示列] リストが更新されて、選択したテンプレートの属性が表示されます。これにより、以前に挿入された属性はすべて削除されます。</p>
[エレメント名]	一致対象のイベントにより参照される PI AF エレメントです。ワイルドカード文字を使用して、名前の一部を指定できます。PI AF Server バージョン 2.8 以降を使用している場合は、特定のエレメントのパスを指定できます。
[エレメントテンプレート]	<p>一致対象のイベントにより参照されるエレメントのエレメントテンプレートです。ベースエレメントテンプレートを選択すると、派生したテンプレートからのエレメントを参照するイベントが含まれることに注意してください。</p> <p>PI AF Server バージョン 2.6 以降では、エレメントテンプレートでフィルタ一処理する必要があります。</p>
[イベントカテゴリ]	一致対象のイベントのカテゴリです。
[最小期間]	一致対象のイベントの最小期間です。値と時間単位の略号を指定します。
[最大期間]	一致対象のイベントの最大期間です。値と時間単位の略号を指定します。

入力	説明
[検索モード]	<p>検索開始と検索終了により指定された期間に相対して一致イベントを検索するために使用される方法です。</p> <ul style="list-style-type: none">active in range 指定した期間内の任意の時間にアクティブであったイベントを見つけてます。entirely in range 指定した期間内に開始および終了したイベントを見つけてます。starting in range 指定した期間内に開始し、期間内または期間後に終了したイベントを見つけてます。ending in range 指定した期間内に終了し、期間前または期間内に開始したイベントを見つけてます。in progress 指定した期間内に開始し、期間内に終了しなかったイベントを見つけてます。PI AF Server バージョン 2.6 以降でのみ利用できます。
[並べ替え順序]	<p>返却イベントを並び替えるために関数が使用する方法。</p> <ul style="list-style-type: none">name ascending イベントをイベント名の昇順(A から Z)で並べます。name descending イベントをイベント名の降順(Z から A)で並べます。start time ascending イベントを開始時刻の古いものから新しいものの順に並べます。start time descending イベントを開始時刻の新しいものから古いものの順に並べます。end time ascending イベントを終了時刻の古いものから新しいものの順に並べます。end time descending イベントを終了時刻の新しいものから古いものの順に並べます。
[重大度]	一致対象のイベントの重要度です。演算子と重要度のタイプを指定します。指定された条件に基づいて一致対象イベントがフィルタリングされます。重要度のタイプは値に対応しており、重要度の高い方から順に表示されます。たとえば、「< Warning」と入力した場合、重要度が Warning 未満のイベント (Information または None) が検索されます。

入力	説明
[属性値フィルター]	<p>一致対象のイベントをフィルター処理するために使用できる、最大 4 つの属性条件です。</p> <p><u>注: 属性値フィルターを指定する前に、イベントテンプレートを指定する必要があります。</u></p> <p>各フィルターに対して、次を指定します。</p> <ul style="list-style-type: none">属性一致対象のイベントをフィルター処理するために使用されるイベント属性です。利用できる属性は、選択したイベントテンプレートにより異なります。演算子指定した属性値に適用される関係演算子です。利用できる演算子は、属性のデータタイプにより異なります。値一致する属性を検索するために使用される値です。たとえば、【演算子】フィールドを[=]に設定した場合、イベントは、指定した属性がこの値に等しいものに制限されます。
[アックフィルター]	<p>一致対象イベントをアック可能なイベントに制限します。</p> <ul style="list-style-type: none">アック済みのイベントを検索するには、[アック済]チェックボックスをオンにします。アックされていないイベントを検索するには、[未アック]チェックボックスをオンにします。現在のアック状態に関係なく、アック可能なすべてのイベントを検索するには、両方のチェックボックスをオンにします。
[注釈付きに制限]	注釈付きのイベントのみを検索するには、このチェックボックスをオンにします。注釈付きかどうかに関係なく、すべてのイベントを検索するには、このチェックボックスをオフにします。

入力	説明
[表示列]	<p>返される関数配列内の列。リストには、属性の名前が含まれます。既定では、すべてのイベントに対して生成される仮想属性と選択したイベントテンプレートからのイベント属性が、リストに含まれます。</p> <p>関数は属性のパスを含めます。つまり、関数は、名前と階層内の位置により、属性を一意に識別します。次の操作を実行できます。</p> <ul style="list-style-type: none">リストされているすべての属性を、返される関数配列の列として含めるには、[すべて選択]チェックボックスを選択します。特定の属性を含めるには、対応するチェックボックスを選択します。特定の属性を、返される関数配列内の列としては含めない場合は、対応するチェックボックスをオフにします。をクリックして[属性の追加]ウィンドウを開きます。このウィンドウでは、返される関数配列内の列として追加する属性を選択できます。「子イベント属性を列として[イベントの比較]タスクペインに追加」を参照してください。属性を右クリックし、次に[親イベントのクローン]をクリックすることにより、親イベントの属性を関数配列の列として挿入できます。「親イベント属性を列として[イベントの比較]タスクペインに追加」を参照してください。リストの下部にある空白のチェックボックスの隣に、イベント属性の名前を入力します。属性を右クリックし、[属性の追加]をクリックして、選択した属性の上に空白の属性を挿入します。次に、イベント属性の名前を入力します。属性を選択し、をクリックして、属性をリストの上位に移動します。属性を選択し、をクリックして、属性をリストの下位に移動します。属性を選択し、をクリックして、属性をリストから削除します。属性を右クリックし、次に[属性の削除]をクリックして、属性をリストから削除します。 <p>選択した列名の長さが 759 文字を超えると、関数は列を個別に処理できなくなります。この場合は、タスクペインに、すべての列をグループとして指定するよう求めるメッセージが表示されます。列をグループとして指定すると、リストには[すべてのイベント属性とテンプレート属性]と表示され、返される関数配列には、デフォルトの属性とその子の属性がすべて含まれますが、ユーザーが指定した追加の属性は含まれません。</p>

入力	説明
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを【表示セル】に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を【表示セル】に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>

次の

[ワークシートのイベント](#)

[子イベント属性を列として\[イベントの比較\]タスクペインに追加](#)

[Excel の PI DataLink 設定を管理する](#)

イベントの比較関数のパス注釈

イベントの比較関数は属性をパスで識別します。これにより、属性名には常にパス情報を含める必要があり、各行に表示された一致したイベントに対して指定します。属性を指定する際に、注釈を使用してイベント階層内の適切な位置を指定します。

サポートされるパス注釈

注釈	説明
. A1	一致したイベントの A1 属性。
.. A1	一致したイベントの親イベントの A1 属性。
..\.. A1	一致したイベントの祖父イベントの A1 属性。
.\E1 A1	一致したイベントの E1 子イベントの A1 属性。

注釈を結合することにより、さらに上の階層で属性を指定できます。たとえば、階層内の異なるイベントに **Duration** 属性を指定できます。

- 曾祖父イベント: ..\..|..|Duration
- 一致したイベント: ..|Duration
- Phase1 という名前の子イベント: .\Phase1|Duration

イベントの比較の例

イベントの比較関数を使用すると、書式で容易に親イベントを比較でき、子イベントから属性を確認できます。複数の温度警告イベントがあり、各イベントにはポンプの温度が一定のレベルに達すると温度が危険な状態にあることを示す单一の子イベントがあると仮定します。各温度警告イベント用に格納された温度危険度イベントに関して記録された開始時刻、間隔、および温度を示す関数配列を取得できます。親イベントの名前は、**TempAlert** で始まり、その子イベントは、**TempThreat** という名前を持ちま

す。PI AF Server AFSRV1 上の Production データベースに過去 1 週間に格納されたイベントの関数配列を取得するには、以下の引数を入力します。

入力	値
データベース	\\AFSRV1\\Production
検索 開始	*-1w
検索 終了	*
イベント名	TempAlert*
列見出し	<p>以下のいずれかを選択します:</p> <ul style="list-style-type: none"> • Event name • . Start time • . End time • . Duration • . Primary element • .\TempThreat Start time* • .\TempThreat Duration* • .\TempThreat Temperature* <p>*[属性の追加] ウィンドウからリストに追加します。</p>

関数は次の配列を返します。

Event name	. Start time	. End time	. Duration	. Primary element	.\TempThreat Start time	.\TempThreat Duration	.\TempThreat Temperature
TempAlert-001	19-Aug-13 05:11:00	19-Aug-13 09:12:23	0 4:01:23	Pump5	19-Aug-13 06:12:00	0 4:01:23	19-Aug-13 06:12:00
TempAlert-002	19-Aug-13 06:17:00	19-Aug-13 09:52:00	0 3:35:00	Pump6	19-Aug-13 05:48:00	0 3:35:00	19-Aug-13 05:48:00
TempAlert-003	19-Aug-13 12:27:00	19-Aug-13 14:28:46	0 2:01:46	Pump5	19-Aug-13 12:44:00	0 2:01:46	19-Aug-13 12:44:00

アセットフィルター検索関数

アセットフィルター検索関数は、指定した検索条件に一致するアセット(PI AF エレメントまたは PI AF 属性)を取得します。検索条件にはエレメント名、エレメントテンプレート、エレメントカテゴリ、エレメントの説明、および属性の値を含めることができます。返されたアセットを静的な値、または関数配列としてワークシートに挿入できます。

[アセットフィルター検索] タスクペインの一番上にあるフィールドで、取得する PI AF エレメントを指定します。タスクペインの下にあるフィールドで、取得する属性、出力フォーマット、および位置を指定します。

詳細については、以下のトピックを参照してください:

- [検索](#)

検索ツールまたはアセットフィルター検索関数のいずれかを使用して、データアイテムを検索できます。

- [フィルタリングによるアセットの検索](#)

この手順に従って、アセットフィルター検索関数を使用して、返すエレメント、およびワークシートの出力内容を指定します。

- [Excel の PI DataLink 設定を管理する](#)

[設定] ウィンドウを使用して、アセットフィルター検索関数が取得するエレメント、および属性の最大数を設定します。

プロパティ関数

プロパティ関数は、指定されたデータアイテムのプロパティ値を返します。

入力	説明
ルートパス	指定データアイテムへの共有パス。有効な項目は以下のものを含みます: <ul style="list-style-type: none">• PI Data Archive サーバー名 (データアイテムが PI ポイントの場合)。• PI AF Server とデータベース (データアイテムが PI AF 属性の場合)。• ブランク (データアイテムが既定の PI Data Archive サーバーにある PI ポイントの場合)。 <p>「データアイテム」を参照してください。</p>
データアイテム	関数がプロパティ値を取得する PI ポイント名または PI AF 属性名(複数指定可)。 PI ポイント名または PI AF 属性名を含むセル範囲を参照し、それぞれに対する値を表示します。既定では、参照データアイテムが列の場合、PI DataLink は行に値を書き込み、参照データアイテムが行の場合は、PI DataLink は列にデータを書き込みます。

入力	説明
プロパティ	<p>関数が値を取得するプロパティ。リストされるプロパティは、入力したデータアイテムにより異なります。</p> <ul style="list-style-type: none"> PI ポイント単体の場合:リストには、入力したポイントのポイントクラスから取得されたポイント属性が含まれます。 <p><u>注:</u> このリストでは、uom を <i>EngUnits</i> の代わりに使用します。ポイント属性 <i>EngUnits</i> の値を確認したい場合は、uom を選択します。</p> <ul style="list-style-type: none"> PI AF 属性単体の場合:リストには、4つの PI AF 属性プロパティ (categories、description、uom、および piPoint)が含まれます。 複数セルへの参照の場合:リストは、最初の参照されるセルにあるデータアイテムのタイプにより異なります。 <ul style="list-style-type: none"> PI ポイントの場合:リストには、対象ポイントの <i>PointClass</i> から取得したポイント属性が含まれます。 PI AF 属性の場合:リストには、4つの PI AF 属性プロパティ (categories、description、uom、および piPoint)が含まれます。 PI DataLink が対象のデータアイテムを見つけられないと、リストはブランクになります。 <p>対象のプロパティを選択するか、プロパティを含むセルへの参照を指定します。</p>
表示セル	<p>関数が取得した関数配列を書き込むワークシート上のセル。関数タスクペインを開く前にセルを選択した場合、PI DataLink は選択したセルを[表示セル]に入力します。</p> <p>関数配列の左上隅を指定します。PI DataLink では、指定されたデータを返すために、必要に応じて表示セルの範囲が下方向と右方向に拡張されます。PI DataLink は隣接するセルを上書きすることができます。</p> <p>複数セル配列を[表示セル]に指定していて、その配列より結果の関数配列の方が大きい場合には、PI DataLink は関数式を余ったセルに貼り付けます。</p>

次の

[PITagAtt\(\)](#)

プロパティ関数の例

ワークシートに入力した複数 PI ポイントの属性を確認するには、プロパティ関数に次の引数を指定します。

入力	値
データアイテム	B3..B5 (ポイントを含むセル配列)
プロパティ	説明

関数は次の配列を返します。

BA:CONC.1	Concentration Reactor 1
BA:LEVEL.1	Level Reactor 1
BA:TEMP.1	Temperature Reactor 1

この場合、ポイント名は左側の列に表示されます。さらにプロパティ関数を使用することで、続きの列に属性を表示できます。

章 7

高度なトピック

このトピックは、PI DataLink 関数で作業する際に役立つ、より詳しい情報を提供します。

PI 時間

PI 時間という特別な書式を使用して、時刻や期間を指定できます。PI 時間は固有の省略形を使用しており、それらを組み合わせて時間表記を作成します。

PI 時間の省略形

PI 時間の指定時には、特定の省略形を使用することで、特定の時間単位や参照時間を表すことができます。

時間単位の略号

略号	標準表記	複数形	対応する時間単位
s	second	seconds	秒
m	minute	minutes	分
h	hour	hours	時
d	day	days	日
mo	month	months	月
y	year	years	年
w	week	weeks	週

時間単位を指定するには、略号(s)、標準表記(second)、時間単位の複数形(seconds)を使用できます。有効な値と時間単位を指定する必要があります。秒、分、時間の指定時には、小数値(例: 1.25h)を使用できます。他の時間単位では、小数値の指定はできません。

相対時間の略号

略号	標準表記	対応する基準時間
*		現在時刻

略号	標準表記	対応する基準時間
t	today	本日の 00:00:00 (午前 0 時)
y	yesterday	前日の 00:00:00 (午前 0 時)
曜日の最初の 3 文字。例： sun	sunday	直近の日曜日の 00:00:00 (午前 0 時)
月の最初の 3 文字。例： jun	june	本年 6 月本日の 00:00:00 (午前 0 時)
dec DD	december DD	本年 12 月 DD 日の 00:00:00 (午前 0 時)
YYYY		YYYY 年本月本日の 00:00:00 (午前 0 時)
M-D または M/D		本年 M 月 D 日の 00:00:00 (午前 0 時)
DD		本月 DD 日の 00:00:00 (午前 0 時)

PI 時間表記

PI 時間表記は、絶対時刻、相対時間の省略形、時間オフセットを含むことができます。時間オフセットでは、オフセットの方向 (+ または - のいずれか) とオフセット量 (値と時間単位の省略形) を指定します。

PI 時間表記は、絶対時刻、相対時間の省略形、時間オフセットを含むことができます。時間オフセットでは、オフセットの方向 (+ または - のいずれか) とオフセット量 (値と時間単位の省略形) を指定します。

ストラクチャ	例
絶対時刻のみ	24-aug-2012 09:50:00
相対時間の省略形のみ	t
時間オフセットのみ	+3h
時間オフセットを伴う相対時間の省略形	t+3h

たとえば、PI 時間表記は以下のようなストラクチャを持つことができます。

タイムスタンプ指定

タイムスタンプに対する入力の指定では、以下のものを含む時間表記を入力できます。

- 固定時間

絶対時刻は、現在時刻に関係なく常に同じ時間を表します。

入力	内容
23-aug-12 15:00:00	2012 年 8 月 23 日午後 3:00

入力	内容
25-sep-12	2012 年 9 月 25 日の午前 0 時

- 相対時刻の略号

相対時間の略号は、現在時刻を起点にした時間を表します。

入力	内容
*	現在時刻 (今)
3-1 または 3/1	本年 3 月 1 日の 00:00:00 (午前 0 時)
2011	2011 年本月本日の 00:00:00 (午前 0 時)
25	今月 25 日の 00:00:00 (午前 0 時)
t	本日(今日)の 00:00:00 (午前 0 時)
y	前日(昨日)の 00:00:00 (午前 0 時)
tue	直近の火曜日の 00:00:00 (午前 0 時)

- 時間オフセットを伴う相対時刻の略号

時間オフセットを相対時刻の略号と一緒に使用した場合、指定した時刻に対して時間オフセットが加算または減算されます。

入力	内容
*-1h	1 時間前
t+8h	今日の 08:00:00 (午前 8:00)
y-8h	一昨日の 16:00:00 (午後 4:00)
mon+14.5h	直近の月曜日の 14:30:00 (午後 2:30)
sat-1m	直近の金曜日の 23:59:00 (午後 11:59)

- 時間オフセット

時間オフセットのみが入力された場合、暗黙の相対時刻を起点とした時間が指定されます。暗黙の相対時刻は、式を入力した場所に応じて、現在時刻または別の時刻になります。

入力	内容
-1d	現在時刻の 1 日前
+6h	現在時刻の 6 時間後

時間間隔の仕様

時間間隔を入力することにより、ある期間における値の収集間隔や計算間隔を設定できます。たとえば、時間間隔を 60 分とした場合、12 時間の間の 1 時間毎の平均を計算します。時間間隔を指定するには、有効な値と時間単位を入力します。

- 正の値は、指定期間内の出来るだけ早い時刻に開始し、出来るだけ遅い時刻またはそれ以前の時刻に終了する間隔を設定します。

開始時刻	2:00:00
終了時刻	3:15:00
時間間隔	30m
返される間隔	2:00:00 - 2:30:00 2:30:00 - 3:00:00

- 負の値は、指定期間内により遅い時刻で終わり、より早い時刻またはそれより後に始まる間隔の定義を行います。

開始時刻	2:00:00
終了時刻	3:15:00
時間間隔	-30m
返される間隔	2:15:00 - 2:45:00 2:45:00 - 3:15:00

式

PI DataLink における式とは、関数で使用可能なパフォーマンスイクエージョンであり、PI System データアイテムに対する数学的な演算や計算を組み込むことができます。たとえば、PI Data Archive から値を取得して計算するために、PI DataLink 関数内で式を使用できます。

注: PI DataLink では、式に組み込むことができるデータアイテムに対する制限もあります。「[式のデータアイテムに関する制限](#)」を参照してください。

一部の PI DataLink 関数では、入力値としてデータアイテムか式のいずれかを指定します。こうした関数の場合、[データアイテム] および [式] オプションがタスクペインの一番上に表示されます。また、[Filter Expression] フィールドを持つ PI DataLink 関数もあります。フィルター式を指定すると、関数が返す値を制限できます。

式を使用可能な PI DataLink 関数には、以下のものがあります:

- [ヒストリ値関数](#)
- [圧縮データ関数](#)
- [定間隔圧縮データ関数](#)
- [時間指定圧縮データ関数](#)
- [演算データ関数](#)
- [条件指定時間データ関数](#)

次のセクションでは、式に関する基本事項を説明します。PI DataLink は、PI Data Archive でサポートされている関数構文、演算子、関数にすべて対応しています。

パフォーマンスイクエージョンと例について詳しくは、PI Server のトピック 「Performance Equations syntax and functions reference (パフォーマンスイクエージョンの構文と関数の参照)」 を参照してください。

PI DataLink 関数での式の使用

PI DataLink 関数で式を使用すると、PI System のデータアイテムを基にした数学的 operation や演算を組み込むことができます。データアイテムや式を使用可能な関数では、タスクペインの一番上に [Data item] と [Expression] オプションが表示されます。

1. 関数タスクペインを開きます。

2. 関数タスクペインの一番上にある [Expression] オプションをクリックします。

[Data item(s)] フィールドが [Expression(s)] フィールドに変わります。他のフィールドの有効/無効も変更されることがあります。

3. [Expression(s)] フィールドに直接、式を入力するか、完全な式を含むセルへの参照を入力します。

次の例では、セル参照を使用する必要があります。

- 関数で複数の式を入力するには

- [Expression(s)] フィールドに、完全な式を含むセル範囲を入力します。

- 他のセルの参照に使用するアンパサンドなどを含む Excel 関数または Excel 計算演算子を含む式を入力するには

- Excel の数式としてセルにこれらの式を入力します。

注: セル参照で入力された式は、簡単に表示や編集を行えます。セル参照を使用して式を入力しておくと、将来、ワークシートのメンテナンスが容易になります。

式の構文

式を入力する際は、次のガイドラインに従ってください:

- 一重引用符で任意の PI ポイントまたは PI AF 属性名を囲みます。

- 'sinusoid' >1

- 一重引用符で任意の時間式を囲みます。

- 't'

- '11-Apr-17'

- 二重引用符で任意の文字列値またはデジタルステートを囲みます。

- "stringtag" = "ACME"

Excel のセル内の式

Excel のセルに式を入力するときは、適切なメソッドを使用します。

- **Excel の文字列**

一重引用符を 1 つ入力してから、式を続けます。この結果、PI ポイントまたは PI AF 属性名で始まる式では、2 つの一重引用符が連続することになります。

```
'abs('sinusoid')>1  
"sinusoid">1
```

Excel では一重引用符があると、式が文字列として解釈されます。

式に Excel の関数や計算演算子（式でセル参照に使用される場合を含む）が含まれない場合にのみ、この方法を使用します。

- **Excel の数式**

等号を 1 つ入力してから、二重引用符で囲んだ式を続けます。

```
=" abs('sinusoid') > 1"  
=" 'sinusoid' > 1"
```

式に Excel の関数や計算演算子（式でセル参照に使用されるアンパサンドなど）が含まれる場合は、この方法を使用します。

この方法は、あらゆる式で使用できます。

式でのセル参照

式にセル参照を組み込むには、Excel のセルで式を指定する必要があります（タスクペインのフィールドでは、フィールド全体に対するエントリを指定するセル参照のみを使用できます）。

Excel のセルに、このアプローチに沿って Excel の数式として式を入力します。

- セル参照の前と後の文字列に式を分割します。
- アンパサンド (&) の計算演算子を持つすべての文字列を連結します。Excel の計算では、アンパサンドの前後の文字列が連結され、1 つの文字列となります。

例

- **単一値のセル参照**

セル B6 で指定された PI ポイントが 0 より大きい場合に TRUE を返す式が必要な場合、

```
'Point in B6' > 0
```

式を格納するセルで、次のコマンドを入力します。

```
="" & B6 & "' > 0";
```

Excel ではセル B6 の値（ポイント sinusoid など）に基づいて、数式が計算されます。PI DataLink 関数では、ブール式を必要とするすべての式フィールドで、このセルを参照できます。

```
'sinusoid'>0
```

- **パフォーマンスイクエージョン関数への複数のセル参照の入力**

Excel ワークシートからの入力で、TimeGT パフォーマンスイクエージョン関数を使用する式が必要だとします。

6	ポイント	sinusoid
7	開始時刻	t
8	終了時刻	*
9	最小	40

この関数では、PI ポイントが特定の値を超えている時間の合計が算出されます。このワークシートでは、セル B6 にポイント、B7 に開始時刻、B8 に終了時間、B9 に値が含まれています。開始時刻と終了時刻は、PI 時間の式として入力されます。対応する式は次のとおりです。

```
TimeGT('Point in B6','Time in B7','Time in B8',Value in B9)
```

式を格納するセルで、次のコマンドを入力します。

```
= "TimeGT('" & B6 & "','" & B7 & "','" & B8 & "','" & B9 & ")";
```

Excel では、セルの値に基づいて、数式が計算されます。

```
TimeGT('sinusoid','t','*',40)
```

- パフォーマンスイクエージョン関数への Excel 関数とセル参照の入力

時間の入力に PI 以外の時刻形式が使用されている場合に、同じパフォーマンスイクエージョン関数を使用するとします。

14	ポイント	sinusoid
15	開始時刻	4/16/2017 12:00:00 AM
16	終了時刻	4/17/2017 12:00:00 AM
17	最小	40

この場合、時間は、特定書式の文字列として入力されます。Excel の TEXT 関数を使用すると、時間文字列を日付に変換できます。対応する式は次のとおりです。

```
TimeGT('Point in B14','TEXT(B15,"DD-MMM-YYYY HH:MM:SS")','TEXT(B16,"DD-MMM-YYYY HH:MM:SS")',Value in B17)
```

式を格納するセルで、次のコマンドを入力します。

```
= "TimeGT('" & B14 & "','" & TEXT(B15,"DD-MMM-YYYY HH:MM:SS") & "','" & TEXT(B16,"DD-MMM-YYYY HH:MM:SS") & "','" & B17 & ")";
```

Excel では、セルの値に基づいて、数式が計算されます。

```
TimeGT('sinusoid','16-Apr-2017 00:00:00','17-Apr-2017 00:00:00',40)
```

式のデータアイテムに関する制限

PI DataLink では、式の中で以下のようなデータアイテムを使用できます:

- PI ポイント
- PI ポイントへのデータ参照を格納した PI AF 属性
- 固定値を格納した PI AF 属性

さらに、PI DataLink 関数は、1 つの PI Data Archive Server からデータのみを返すことができます。このため、1 つの式内の PI ポイントは、PI AF 属性により参照されるものを含め、同じ PI Data Archive Server 上に格納されている必要があります。同様に、フィルター式内の PI ポイントを格納する PI Data Archive Server は、[データアイテム] や [式] フィールドで定義された PI ポイントと同じである必要があります。

式の例

次の例では、データアイテムの評価や、データアイテムの値の操作または計算を行う式を示しています。

- **PI ポイントの演算操作**

```
('sinusoid')^3 + 'cdf144'/10
```

PI ポイント `sinusoid` の値の 3 乗と、PI ポイント `cdf144` の値を 10 で除算した合計が返されます。

- **PI ポイントのブール評価**

```
abs('mytag') >= 14.65
```

PI ポイント `mytag` の絶対値が 14.65 以上の場合、`true` (ゼロ以外) が返されます。

- **属性の PI AF ブール評価**

```
'\\Server\Database\Element|Manufacturer' = "ACME"
```

PI AF 属性 `Manufacturer` の値が `ACME` の場合、`true` が返されます。

- **2 つの PI ポイントの同時条件に沿った複雑なブール評価**

```
'sinusoid' < 45 と sqr('vdf1002') > 2
```

PI ポイント `sinusoid` の値が 45 よりも小さく、PI ポイント `vdf1002` の値の平方根が 2 よりも大きい場合に、`true` が返されます。

- **デジタルステートポイントの評価**

```
StateNo('BA:Phase.1')
```

デジタルステートポイントの値は返されず、デジタルポイント `BA.Phase.1` のデジタルステート数（「デジタルステートコード」）が返されます。

フィルター式

PI DataLink 関数でフィルター式を使用すると、記録した値をブール型のパフォーマンスイクエージョンでフィルターできます。PI DataLink では、式で `false` と評価されたデータが除外されます。

PI DataLink は、PI System から取得した生データにフィルター式を適用します(演算結果の値ではありません)。たとえば、演算データ関数にフィルター式 '`sinusoid` < 70' を追加すると、計算結果から 70 以上の値がすべて除外されます。

すべての有効なブール型のパフォーマンスイクエージョンをフィルター式として使用できます。ただし、フィルター式内のデータアイテムは PI ポイントを参照しなければなりません。複雑な式を構成することもできます。たとえば、フィルター式を使用して、記録したデータ値の中から例外的なピーク値を除外できます。

[フィルター済みの表示] チェックボックスを選択すると、フィルター式により結果から除外された値の代わりに `Filtered` が表示されます。

関数の手入力

PI DataLink から PI DataLink 関数を構築できますが、使い慣れたユーザーは Excel の数式バーに関数を直接入力する場合があります。

以下では、数式バーに直接入力された PI DataLink 関数に関するトピックを扱います。

関数の手動定義

関数タスクスペインを使用したくない場合は、PI DataLink 関数を手動で定義できます。

1. 想定する取得値の数に基づき、出力配列用に適切な大きさのセル範囲をワークシートで選択します。
2. Excel の数式バーに PI DataLink 関数とその引数を入力します。
3. **Ctrl + Shift + Enter** キーを押して、選択した表示セルに PI DataLink 関数を入力します。

配列数式の詳細については、Microsoft Excel のオンラインヘルプを参照してください。

引数の手動入力に関するガイドライン

次の表には、PI DataLink 関数に引数として入力するデータのタイプと、そのデータを入力する際のガイドラインを記載しています。

引数として入力するデータ	ガイドライン
標準関数	式を使用可能な引数に、様々な関数を指定できます。関数のリストについては、PI Server のトピック「 Built-in performance equation functions (内蔵のパフォーマンスイクエージョン関数)」を参照してください。
文字列	文字列の引数は二重引用符で囲みます。たとえば、 <code>casaba</code> という名前の PI Data Archive Server から、PI ポイント <code>sinusoid</code> の現在値をタイムスタンプ付きで取得するには、 1×2 の出力配列を選択して、次のように入力します。 <code>=PICurrVal("sinusoid", 1, "casaba")</code>
式	二重引用符で式の引数を囲みます。式の中では、PI ポイントや PI AF 属性名を一重引用符、文字列値やデジタルステートを二重引用符で囲みます。 たとえば、PI ポイント <code>cdm158</code> が <code>Manual</code> と等しいとき、値 <code>true</code> を返す式の場合、次のように入力します。 <code>""cdm158""=""Manual""</code>
Cell references	PI DataLink 関数の引数として、セル参照を使用できます。たとえば、ワークシートセルに次の値があると仮定します。 <ul style="list-style-type: none">• A1: <code>"sinusoid"</code>• A2: 1• A3: <code>"casaba"</code> 次に、数式バーで次の関数を入力します。 <code>=PICurrVal(A1, A2, A3)</code> これは、次の入力と同等です。 <code>=PICurrVal("sinusoid", 1, "casaba")</code>

引数として入力するデータ	ガイドライン
ルートパス	<p>入力値は二重引用符で囲みます。たとえば、セル B3 から B5 で指定される PI AF 属性の現在値を、PI AF Server 「DLAFPI」 にあるデータベース 「MyTest」 のエレメント 「Reactor」 から取得するには、次のように入力します。</p> <p><code>=PICurrVal(B3:B5,0,"\\DLAFPI\\MyTest\\Reactor")</code></p> <p>データアイテムがデフォルトの PI Data Archive サーバー上の PI ポイントである場合など、ルートパスを指定しないときは、2 つの二重引用符を入力します。</p> <p>""</p> <p>有効なエントリの詳細については、データアイテムを参照してください。</p>
表示コード	関数がどの追加データを取得するか、関数がどの向きに出力するかを決定します。「 表示コード 」を参照してください。

表示コード

PI DataLink 関数構文は、整数の *OutCode* 引数を含みます。この引数に指定した表示コードにより、関数が取得する追加データと、表示セルの方向が決定されます。

PI DataLink タスクペインは表示コードの値を自動生成します。ただし、Excel の数式バーに手動で関数を入力した場合、適切な表示コードの値を計算して指定する必要があります。

表示コードは、2 進数のビット文字列を整数で表現したものです。PI DataLink では、ビットの意味は関数によって異なります。

現在値関数と圧縮データ関数におけるビットの意味

ビット	目的
第 1	値の左側の列にタイムスタンプを表示します
第 2	値の上の行にタイムスタンプを表示します

他の PI DataLink 関数におけるビットの意味

ビット	目的
第 1	タイムスタンプの表示
第 2	列ではなく行でデータを表示する
第 3	有効データ割合の表示
第 4	値の個数の非表示
第 5	拡張ステータスの表示
第 6	注釈の表示

ビット	目的
第 7 ¹	開始時刻の表示
第 8 ¹	終了時刻の表示
第 9 ¹	最小/最大時刻の表示

¹ 第 1 ビットも有効にする必要があります。

OutCode 引数を「0」にすると、関数は値を列で返します(追加データはなし)。*OutCode* 引数を計算するには、次の数式を使用します。

$$\text{OutCode} = \sum_i 2^{x_i - 1}$$

where x is the set of enabled bits

たとえば、*OutCode* 引数を「0」にすると、*PISampDat()*関数は、定間隔ヒストリデータを指定した表示セルに返します。*OutCode* を「1」にすると、関数は、 $n \times 2$ の配列の 1 列目にタイムスタンプを、2 列目に定間隔ヒストリデータを返します。*OutCode* 引数を「3」に設定すると、関数は $2 \times n$ 配列の 1 行目にタイムスタンプを、2 行目に定間隔ヒストリデータを出力します。ここで、 n は定間隔ヒストリデータの個数を意味します。

注: サポートされているビットや *OutCode* は、関数によって異なります。ある関数で有効な値であっても、他の関数では有効でないことがあります。各関数がその *OutCode* でサポートするビットについては、[関数リファレンス](#)を参照してください。

例

演算データ関数で、有効データ割合、開始時刻、最小/最大時刻を表示する必要があると仮定します。*show-time-stamps* ビットを有効にすると、開始時刻、最小/最大時刻が表示されます。*OutCode* 引数の計算手順:

Enabled bits = {Show time stamps, Show percent good, Show start time, Show min/max time}
 $x = \{ 1, 3, 7, 9 \}$

$$\begin{aligned} \text{OutCode} &= \sum_i 2^{x_i - 1} \\ &= 2^{1-1} + 2^{3-1} + 2^{7-1} + 2^{9-1} \\ &= 2^0 + 2^2 + 2^6 + 2^8 \\ &= 1 + 4 + 64 + 256 \\ &= 325 \end{aligned}$$

Excel の数式バーで、次のように入力します。

```
=PIAdvCalcDat("sinusoid","y","t","1h","minimum","time-weighted", 50, 1, 325,"MyDataServer")
```

注: *OutCode* 引数で複数列/行の表示を指定したにもかかわらず、表示されない場合は、関数配列を右クリックして**[DataLink 関数の再計算]**を選択し、関数配列を更新してください。

データ書き出し用の関数

データを PI Data Archive または PI AF から取得する一般的な PI DataLink 関数とは異なり、**PIPutVal()**関数と **PIPutValX()**関数ではワークシートの値が PI Data Archive または PI AF に書き込まれます。

PIPutVal()関数と **PIPutValX()**関数では、特定のタイムスタンプを持つ既存の値をユーザーが入力する新しい値で置き換えることができます。マクロが有効なブックで、**PIPutVal()**関数または **PIPutValX()**関数を実行する必要があります。

PI DataLink と一緒に配布されるブック例を使用して、PI Data Archive または PI AF にデータを書き込むことができます。「[PI Data Archive や PI AF にデータを書き込む](#)」を参照してください。

次の

[PIPutVal\(\) や PIPutValX\(\)](#)

PI Data Archive や PI AF にデータを書き込む

PI Data Archive や PI AF にデータを書き込むとき、**PIPutVal()**関数や **PIPutValX()**関数を使用するには、PI DataLink と一緒に配布されたワークブック例を活用してください。

注: これらの関数を使用する Visual Basic for Application (VBA) コードの記述例を閲覧するには、ブック例を Visual Basic エディタで開き、**PutVal_code** モジュールを確認します。

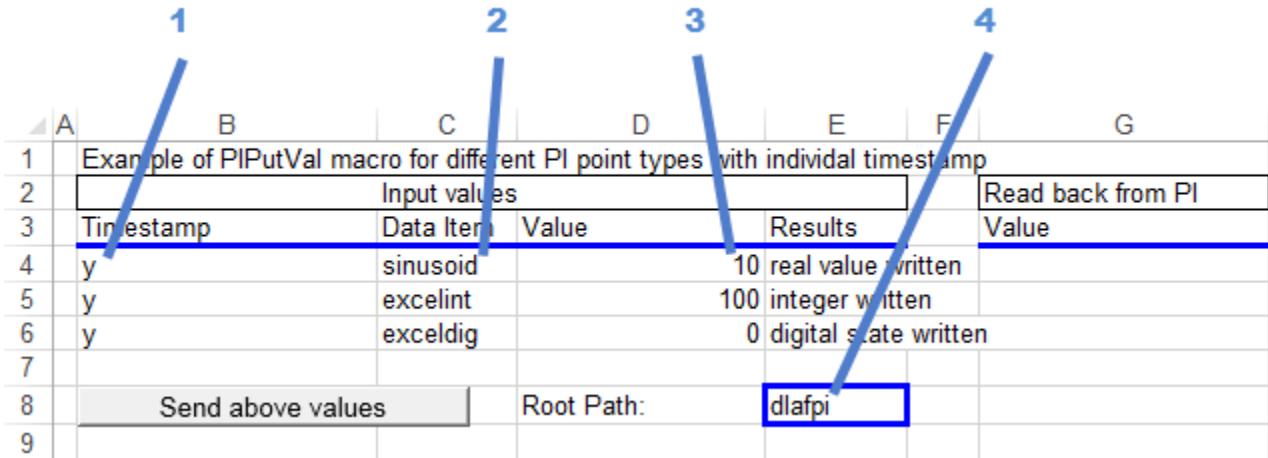
1. **..../PIPC/Excel** ディレクトリでブック例を開きます。

- **piexam32.xls** (32 ビット版 Excel に対応した PI DataLink と一緒に配布)
- **piexam64.xls** (64 ビット版 Excel に対応した PI DataLink と一緒に配布)

ブック例には、1 つのワークシート、**PutVal** が含まれています。このワークシートには、2 つのセクションがあります。最初のセクションでは、それぞれタイムスタンプの値が異なる複数のデータアイテムの値を入力する **PIPutVal** 関数を使用します。2 つめのセクションでは、どれもタイムスタンプの値が同じ複数のデータアイテムの値を入力する **PIPutValX** 関数を使用します。

2. **PutVal** ワークシートの適切なセルで、タイムスタンプとデータアイテム、書き込む値と一緒に、入力するデータアイテムのサーバーを指定するルートパスを入力します。

サーバーに書き込む値が書き込まれたワークシート



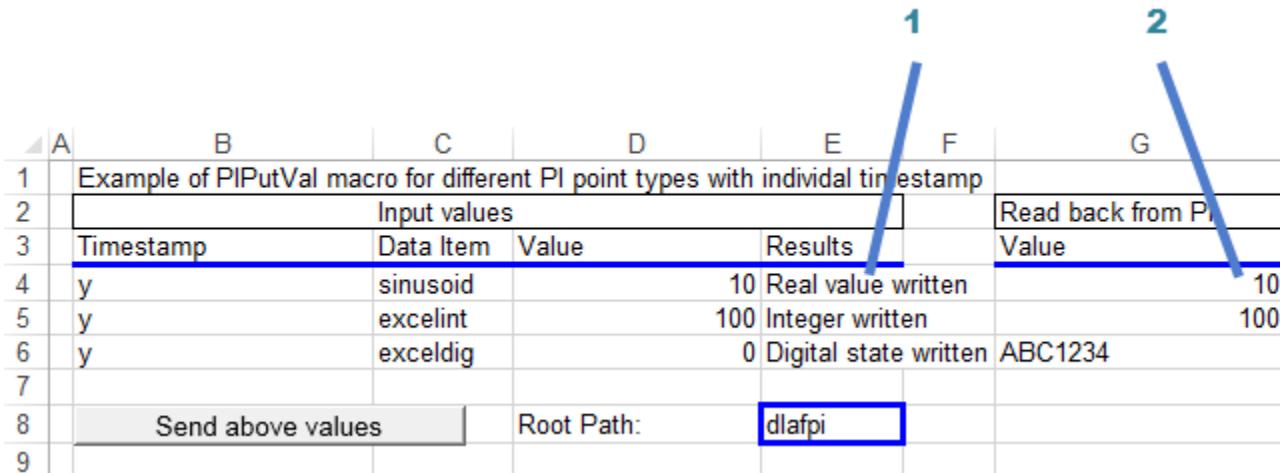
A	B	C	D	E	F	G
1	2	3	4			
Example of PIPutVal macro for different PI point types with individual timestamp						
Input values						
Timestamp	Data Item	Value	Results	Read back from PI	Value	
4	sinusoid		10 real value written			
5	excelint		100 integer written			
6	exceldig		0 digital state written			
7						
8	Send above values		Root Path: dlafpi			
9						

- 1. タイムスタンプ
- 2. データ アイテム
- 3. 書き込む値
- 4. ルート パス

3. [Send above values] (以上の値を送信) ボタンをクリックしてマクロを起動します。

このマクロでは、指定された値を書き込み、応答を記録するために PIPutVal() または PIPutValX() 関数が使用され、次にサーバーに格納された値を返すために PIArcVal 関数が使用されます。

マクロ実行後のワークシート



A	B	C	D	E	F	G
1	2					
Example of PIPutVal macro for different PI point types with individual timestamp						
Input values						
Timestamp	Data Item	Value	Results	Read back from PI	Value	
4	sinusoid		10 Real value written		10	
5	excelint		100 Integer written		100	
6	exceldig		0 Digital state written	ABC1234		
7						
8	Send above values		Root Path: dlafpi			
9						

- 1. PIPutVal 関数の応答
- 2. PIArcVal 関数で取得した値

章 8

関数リファレンス

[PI DataLink]タスクペインから PI DataLink 関数を生成できます([PI DataLink 関数の概要](#)を参照)。入力値に応じて、生成される関数配列は異なります。Excel の数式バーから同じ関数を直接入力することもできます([関数の手入力](#)を参照)。

このセクションは、サポートされている PI DataLink 関数に関する情報を記載しています。各トピックは、関数の構文と引数を説明しています。また、ほとんどのトピックには例が含まれています。以下のセクションでは、種類別に関数が分類されています。

单一値関数

单一値関数は、特定の時間のデータアイテムの値を取得します。单一値関数は、データアイテムごとに1つの値を取得します。

PICurrVal()

PI ポイントや PI AF 属性の現在値や最新値を取得します。未来の PI ポイントの場合、戻り値は、現在の時刻を基準に記録されている値のタイムスタンプによって決定します。

- 記録されている値がすべて、現在の時刻より前の場合、この関数で返される値は、最後に記録された値です。
- 記録されている値のタイムスタンプが、現在の時刻をまたいでいる場合、この関数は、内挿値を返します。
- 記録されている値がすべて、現在の時刻より後の場合、この関数でデータは返されません。

この関数は一括呼び出しをサポートしています。

構文

`PICurrVal(DataItem, OutCode, RootPath)`

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。

引数	タイプ	説明
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1 とビット 2 に対応しています。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PICurrVal("let439",1,"holden")
```

PI ポイント let439 の現在値およびそのタイムスタンプを PI Data Archive Server (名前 holden) から取得し、タイムスタンプを値の左側に表示します。

次の

[現在値関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[大規模データの取得](#)

[内挿値](#)

PIArcVal()

指定したタイムスタンプの PI ポイント値、または PI AF 属性値を返します。この関数は一括呼び出しをサポートしています。

構文

```
PIArcVal(DataItem, TimeStamp, OutCode, RootPath, Mode)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>TimeStamp</i>	文字列	タイムスタンプ。関数は、これに対する値を返します。「 時間入力 」を参照してください。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1 とビット 2 に対応しています。

引数	タイプ	説明
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。
<i>Mode</i>	文字列	関数による値の取得方法: <ul style="list-style-type: none"> • previous • previous only • interpolated • auto • next • next only • exact time

例

```
=PIArcVal("cdep158", "11-dec-92 19:20", 0, "casaba", "interpolated")
```

1992 年 12 月 11 日午後 7 時 20 分の PI ポイント cdep158 の内挿値を、PI Data Archive Server (名前は casaba) から返します。

次の

[ヒストリ値関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

[大規模データの取得](#)

PIExpVal()

指定されたタイムスタンプにおけるパフォーマンスイクエージョンの計算値を返します。

構文

```
PIExpVal(Expression, TimeStamp, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンスイクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>TimeStamp</i>	文字列	タイムスタンプ。関数は、これに対する値を返します。「 時間入力 」を参照してください。

<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1 とビット 2 に対応しています。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PIExpVal("sqr('sinusoid')","y",0,"thevax")
```

PI ポイント `sinusoid` の、昨日の午前 0 時の時点での、PI Data Archive Server (名前 `thevax`) から取得された値の平方根を計算します。

次の

[ヒストリ値関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

PITagAtt()

指定されたデータアイテムのプロパティ値を返します。

構文

```
PITagAtt(DataItem, Property, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>Property</i>	文字列	関数が値を取得する PI ポイント名または PI AF 属性名。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PITagAtt(d1,"uom","")
```

セル D1 で指定されている PI ポイントの工業単位を既定の PI Data Archive Server から取得して返します。

次の

[プロパティ関数](#)

複数値関数

複数値関数は時間範囲に関する PI ポイントや PI AF 属性の値を返します。この時間範囲内には 1 つまたはそれ以上の値が含まれます。

PINCompDat()

指定された開始時刻から、指定された数の PI ポイント値や PI AF 属性を返します。

構文

```
PINCompDat(DataItem, STime, NumVals, OutCode, RootPath, Mode)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それに対する値が返されます。
<i>STime</i>	文字列	関数が値を返す期間の開始時刻。「 時間入力 」を参照してください。
<i>NumVals</i>	整数	関数が取得する値の数。 <i>STime</i> を起点とします (負の値を指定すると、時間を遡って値が取得されます)。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1、2、5、および 6 に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。
<i>Mode</i>	文字列	開始時刻付近のどの値を取得するかを決定するために、関数が使用する方法(境界タイプ): <ul style="list-style-type: none">• <code>inside</code>• <code>outside</code>• <code>interpolated</code>• <code>auto</code>

例

```
=PINCompDat("sinusoid","1:00:00",10,1,"","inside")
```

既定の PI Data Archive Server から、PI ポイント `sinusoid` について、今朝の午前 1 時から始まる 10 個の値と対応するタイムスタンプを取得して返します。境界タイプとして、`inside` が使用されます。

次の

[圧縮データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PINCompFilDat()

指定された開始時刻から、指定された数の PI ポイント値や PI AF 属性にフィルターをかけて返します。

構文

```
PINCompFilDat(DataItem, STime, NumVals, FiltExp, FiltCode, OutCode, RootPath, Mode)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>STime</i>	文字列	関数が値を返す期間の開始時刻。「 時間入力 」を参照してください。
<i>NumVals</i>	整数	関数が取得する値の数。 <i>STime</i> を起点とします(負の値を指定すると、時間を遡って値が取得されます)。
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用する布尔型のパフォーマンスイクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。
<i>FiltCode</i>	整数	フィルターされた値にラベル付けをするかを示すコード: <ul style="list-style-type: none">• 1 <i>FiltExp</i> に基づいて関数が結果から除外した値または値のブロックの位置に、<code>Filtered</code> というラベルが挿入されます。• 0 フィルターされた値に、ラベル付けはされません。

引数	タイプ	説明
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1、2、5、および 6 に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。
<i>Mode</i>	文字列	開始時刻付近のどの値を取得するかを決定するために、関数が使用する方法(境界タイプ): <ul style="list-style-type: none"> • <code>inside</code> • <code>outside</code> • <code>interpolated</code> • <code>auto</code>

例

```
=PINCompFilDat("sinusoid","2:00:00",10,"'cdep158'>38",1,1,"","inside")
```

午前 2 時以降の、既定の PI Data Archive Server から、PI ポイント `sinusoid` の値 10 個と対応するタイムスタンプを取得して返します。ただし、PI ポイント `cdep158` の値が 38 よりも大きい場合のみです。境界タイプとして、`inside` が使用されます。

フィルター条件に当てはまらない期間については、「`Filtered`」というステータスが表示されます。

次の

[圧縮データ関数](#)

[フィルター式](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PICompDat()

指定された期間に格納された PI ポイント値または PI AF 属性値を返します。

構文

```
PICompDat(DataItem, STime, ETime, OutCode, RootPath, Mode)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>STime</i>	文字列	関数が値を返す期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を返す期間の終了時刻。「 時間入力 」を参照してください。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1、ビット 2、ビット 4、ビット 5、ビット 6 に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。
<i>Mode</i>	文字列	開始時刻付近のどの値を取得するかを決定するために、関数が使用する方法(境界タイプ): <ul style="list-style-type: none">• <code>inside</code>• <code>outside</code>• <code>interpolated</code>• <code>auto</code>

例

```
=PICompDat("sinusoid", "1:00:00", "3:00:00", 1, "", "inside")
```

既定の PI Data Archive Server から、PI ポイント `sinusoid` の、今朝の午前 1 時から午前 3 時の期間の値と対応するタイムスタンプを取得して返します。境界タイプとして、`inside` が使用されます。

次の

[圧縮データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PICompFilDat()

指定された期間に格納された PI ポイント値または PI AF 属性値にフィルターをかけて返します。

構文

```
PICompFilDat(DataItem, STime, ETime, FiltExp, FiltCode, OutCode, RootPath, Mode)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>STime</i>	文字列	関数が値を返す期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を返す期間の終了時刻。「 時間入力 」を参照してください。
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用するブール型のパフォーマンスイクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。
<i>FiltCode</i>	整数	フィルターされた値にラベル付けをするかを示すコード: <ul style="list-style-type: none">• 1 FiltExp に基づいて関数が結果から除外した値または値のブロックの位置に、<code>Filtered</code> というラベルが挿入されます。• 0 フィルターされた値に、ラベル付けはされません。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1、ビット 2、ビット 4、ビット 5、ビット 6 に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

引数	タイプ	説明
<i>Mode</i>	文字列	<i>STime</i> または <i>ETime</i> 付近のどの値を取得するかを決定するために、関数が使用する方法(境界タイプ): <ul style="list-style-type: none">• <i>inside</i>• <i>outside</i>• <i>interpolated</i>• <i>auto</i>

例

```
=PICompFilDat("sinusoid","2:00:00","10:00:00",'cdep158'>38',1,1,"","inside")
```

午前 2 時から午前 10 時までの期間で、PI ポイント *sinusoid* の値と対応するタイムスタンプを、既定の PI Data Archive Server から取得して返します。ただし、PI ポイント *cdep158* の値が 38 よりも大きい場合のみです。境界タイプとして、*inside* が使用されます。

フィルター条件に当てはまらない期間については、「*Filtered*」というステータスが表示されます。

次の

[圧縮データ関数](#)

[フィルター式](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PISampDat()

指定された期間に定間隔で格納された PI ポイントまたは PI AF 属性の内挿値を返します。

構文

```
PISampDat(DataItem, STime, ETime, Interval, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>STime</i>	文字列	関数が値を返す期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を返す期間の終了時刻。「 時間入力 」を参照してください。

引数	タイプ	説明
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、15m(15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1 とビット 2 に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PISampDat("sinusoid","y","t","3h",1,"")
```

昨日の午前 0 時から今日の午前 0 時までの期間で、PI ポイント *sinusoid* について、既定の PI Data Archive Server からサンプリングされたデータと対応するタイムスタンプを取得し、3 時間間隔で値をレポートします。

次の

[定間隔圧縮データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PISampFilDat()

指定された期間に格納された PI ポイント値や PI AF 属性値を定間隔で内挿し、フィルタリングした値を返します。

構文

```
PISampFilDat(DataItem, STime, ETime, Interval, FiltExp, FiltCode, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>STime</i>	文字列	関数が値を返す期間の開始時刻。「 時間入力 」を参照してください。

引数	タイプ	説明
<i>ETime</i>	文字列	関数が値を返す期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、15m(15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用するブール型のパフォーマンスイクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。
<i>FiltCode</i>	整数	フィルターされた値にラベル付けをするかを示すコード:
		<ul style="list-style-type: none"> • 1 FiltExp に基づいて関数が結果から除外した値または値のブロックの位置に、<code>Filtered</code> というラベルが挿入されます。 • 0 フィルターされた値に、ラベル付けはされません。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 1 とビット 2 に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PISampFilDat("sinusoid", "11-Jan-1997", "+3h", "1h", A1, 1, 1, "")
```

PI ポイント `sinusoid` の定間隔ヒストリデータを既定の PI Data Archive Server から取得し、返します。関数は、「1997 年 1 月 11 日 午前 0 時」、「1997 年 1 月 11 日 午前 1 時」、「1997 年 1 月 11 日 午前 2 時」、「1997 年 1 月 11 日 午前 3 時」の値を取得します。セル A1 の条件が上記の時間を満たさない場合、関数は、その時間に対し「`Filtered`」というステータスを取得します。関数は、1 行目にタイムスタンプを、2 行目に値を表示します。

次の

[定間隔圧縮データ関数](#)

[フィルター式](#)

[関数の手動定義](#)

引数の手動入力に関するガイドライン内挿値

PIExpDat()

一定期間、定間隔で計算したパフォーマンスイクエージョンの値を返します。

構文

```
PIExpDat(Expression, STime, ETime, Interval, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンスイクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、 15m (15分)と入力すると、対象期間中、15分の間隔で1つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット1とビット2に対応しています。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PIExpDat("sqr('sinusoid')","y","t","1h",1,"thevax")
```

昨日の午前0時から今日の午前0時までの1時間間隔で thevax という PI Data Archive Server から取得した PI ポイント **sinusoid** の平方根を計算します。

関数は、各計算値の左側の列にタイムスタンプを表示します。

次の

[定間隔圧縮データ関数](#)[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)[内挿値](#)

PITimeDat()

指定したタイムスタンプにおける PI ポイントや PI AF 属性の実際のサンプリング値または内挿されたサンプリング値を返します。

構文

```
PITimeDat(DataItem, TimeStamps, RootPath, Mode)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が返されます。
<i>TimeStamps</i>	参照	タイムスタンプを含む 1 つ以上のセルに対する参照。関数はここで指定したタイムスタンプに対する値を取得します。「 時間入力 」を参照してください。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。
<i>Mode</i>	文字列	関数による値の取得方法: <ul style="list-style-type: none">• <code>interpolated</code>• <code>exact time</code>

戻り値

指定したタイムスタンプにおける値。タイムスタンプ参照と同じ方向に表示されます。タイムスタンプ参照が行の場合、関数は値を 1 行に表示します。タイムスタンプ参照が列の場合、関数は値を 1 列に表示します。出力配列の向きと、タイムスタンプ参照の向きは、一致していなければなりません。

例

```
=PITimeDat("sinusoid",b1:b12,"","interpolated")
```

セル B1 から B12 に書かれたタイムスタンプに対応する PI ポイント `sinusoid` の内挿値を既定の PI Data Archive Server から取得します。

この例では、関数は、C1:C12 や B14:B25 といった、横 12 セル分の出力配列を必要とします。

次の

[時間指定圧縮データ関数](#)[関数の手動定義](#)[引数の手動入力に関するガイドライン](#)[内挿値](#)

PITimeExpDat()

指定されたタイムスタンプにおいて計算されたパフォーマンスイクエージョンの値を返します。

構文

```
PITimeExpDat(Expression, TimeStamps, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンスイクエージョン。「式」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>TimeStamps</i>	参照	タイムスタンプを含む 1 つ以上のセルに対する参照。関数はここで指定したタイムスタンプに対する値を取得します。「 時間入力 」を参照してください。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

戻り値

指定したタイムスタンプにおける値。タイムスタンプ参照と同じ方向に表示されます。タイムスタンプ参照が行の場合、関数は値を 1 行に表示します。タイムスタンプ参照が列の場合、関数は値を 1 列に表示します。出力配列の向きと、タイムスタンプ参照の向きは、一致していなければなりません。

例

```
=PITimeExpDat("sqr('sinusoid')",b1:b12,"")
```

B1 から B12 で指定された時間における、PI ポイント sinusoid の値を既定の PI Data Archive Server から取得、その値の平方根を計算します。

この例では、関数は、C1:C12 や B14:B25 といった、横 12 セル分の出力配列を必要とします。

次の

[時間指定圧縮データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

計算関数

計算関数は、指定時間範囲での PI ポイント値、PI AF 属性、またはパフォーマンスイクエージョンから新しい値を計算します。

PIAdvCalcVal()

指定期間における PI ポイントや PI AF 属性の計算値を返します。この関数は一括呼び出しをサポートしています。

構文

```
PIAdvCalcVal(DataItem, STime, ETime, Mode, CalcBasis, MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が計算されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Mode</i>	文字列	関数が実行する計算の種類: <ul style="list-style-type: none">totalminimummaximumstdevrangecountaverage (time-weighted)average (event-weighted)
<i>CalcBasis</i>	文字列	計算方法: <ul style="list-style-type: none">time-weightedevent-weighted
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、1 を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。

引数	タイプ	説明
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット 2、3、および 9 をサポートします(<i>Mode</i> が「minimum」、「maximum」、または「range」に設定されている場合)。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PIAdvCalcVal("sinusoid","y","t","total","time-weighted",50,1,4,"")
```

PI ポイント *sinusoid* の時間加重積算値を、既定の PI Data Archive Server に存在する昨日の午前 0 時から今日の午前 0 時までの値を使用し、計算して返します。

関数は結果を 1 で乗算し、有効データの割合が 50% 以上である場合にのみ、結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。

次の

[演算データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[大規模データの取得](#)

PIAdvCalcFilVal()

指定期間にフィルタリングされた PI ポイント値や PI AF 属性値に計算を施した値を返します。

構文

```
PIAdvCalcFilVal(DataItem, STime, ETime, FiltExp, Mode, CalcBasis, SampMode, SampFreq,  
MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が計算されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。

引数	タイプ	説明
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用するブール型のパフォーマンスイクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。
<i>Mode</i>	文字列	関数が実行する計算の種類:
		<ul style="list-style-type: none"> • <code>total</code> • <code>minimum</code> • <code>maximum</code> • <code>stdev</code> • <code>range</code> • <code>count</code> • <code>average (time-weighted)</code> • <code>average (event-weighted)</code>
<i>CalcBasis</i>	文字列	計算方法:
		<ul style="list-style-type: none"> • <code>time-weighted</code> • <code>event-weighted</code>
<i>SampMode</i>	文字列	<i>FiltExp</i> を評価するタイミングを決定するために使用されるサンプリング方法:
		<ul style="list-style-type: none"> • <code>compressed</code> • <code>interpolated</code>
<i>SampFreq</i>	文字列	<i>FiltExp</i> が、 <i>SampMode</i> が <code>interpolated</code> に設定されている場合に評価される頻度。値と時間単位を入力します。たとえば、時間間隔を <code>10m (10 分)</code> とすると、10 分ごとの内挿値が取得されます。 時間間隔の仕様 を参照してください。
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、 <code>1</code> を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。この関数は、ビット <code>2</code> 、 <code>3</code> 、および <code>9</code> をサポートします(<i>Mode</i> が「 <code>minimum</code> 」、「 <code>maximum</code> 」、または「 <code>range</code> 」に設定されている場合)。

引数	タイプ	説明
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PIAdvCalcFilVal("sinusoid","y","t",'cdm158'="Manual","","total","time-weighted","compressed","10m",50,1,4,"")
```

PI ポイント `sinusoid` の時間加重積算値を、既定の PI Data Archive Server にある値を計算して返します。値の積算は、昨日の午前 0 時から今日の午前 0 時までの期間で、PI ポイント `cdm158` が「Manual」に設定されているときに行われます。

関数は結果を 1 で乗算します。サンプリングモードが `compressed` であるため、関数はサンプリング間隔を無視し、ポイント `sinusoid` のヒストリイベントを使用して、フィルター式をサンプリングします。有効データの割合が 50% 以上である場合にのみ、関数は結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。

次の[演算データ関数](#)[フィルター式](#)[関数の手動定義](#)[引数の手動入力に関するガイドライン](#)[内挿値](#)

PIAdvCalcExpVal()

指定した期間の式に基づく計算結果を返します。

構文

```
PIAdvCalcExpVal(Expression, STime, ETime, Mode, CalcBasis, SampMode, SampFreq, MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンス イクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。

引数	タイプ	説明
<i>Mode</i>	文字列	関数が実行する計算の種類: <ul style="list-style-type: none">• <code>total</code>• <code>minimum</code>• <code>maximum</code>• <code>stdev</code>• <code>range</code>• <code>count</code>• <code>average (time-weighted)</code>• <code>average (event-weighted)</code>
<i>CalcBasis</i>	文字列	計算方法: <ul style="list-style-type: none">• <code>time-weighted</code>• <code>event-weighted</code>
<i>SampMode</i>	文字列	<i>Expression</i> を評価するタイミングを決定するために使用されるサンプリング方法: <ul style="list-style-type: none">• <code>compressed</code>• <code>interpolated</code>
<i>SampFreq</i>	文字列	<i>Expression</i> が、 <i>SampMode</i> が <code>interpolated</code> に設定されている場合に評価される頻度。値と時間単位を入力します。たとえば、時間間隔を <code>10m</code> (10 分) とすると、10 分ごとの内挿値が取得されます。 時間間隔の仕様 を参照してください。
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、 <code>1</code> を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2、3、および 9 をサポートします(<i>Mode</i> が「 <code>minimum</code> 」、「 <code>maximum</code> 」、または「 <code>range</code> 」に設定されている場合)。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PIAdvCalcExpVal("sinusoid+'cdt158'","y","t","total","time-weighted","compressed","10m",50,1,4,"")
```

昨日の午前 0 時から今日の午前 0 時までの値を使用し、デフォルトの PI Data Archive Server に存在する値を計算した式 'sinusoid+'cdt158' の時間加重積算値を計算して返します。

関数は結果を 1 で乗算します。サンプリングモードが **compressed** であるため、関数はサンプリング間隔を無視し、**sinusoid** のヒストリイベントと **cdt158** のヒストリイベントを併用して、フィルター式をサンプリングします。有効データの割合が 50% 以上である場合にのみ、関数は結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。

次の

[演算データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PIAdvCalcExpFilVal()

式に基づく計算とフィルタリングを行った PI ポイント値を返します。

構文

```
PIAdvCalcExpFilVal(Expression, STime, ETime, FiltExp, Mode, CalcBasis, SampMode, SampFreq, MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンス イクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用するブール型のパフォーマンス イクエージョンです。式が false と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。

引数	タイプ	説明
<i>Mode</i>	文字列	関数が実行する計算の種類: <ul style="list-style-type: none">• total• minimum• maximum• stdev• range• count• average (time-weighted)• average (event-weighted)
<i>CalcBasis</i>	文字列	計算方法: <ul style="list-style-type: none">• time-weighted• event-weighted
<i>SampMode</i>	文字列	<i>Expression</i> と <i>FiltExp</i> 評価するタイミングを決定するために使用されるサンプリング方法: <ul style="list-style-type: none">• compressed• interpolated
<i>SampFreq</i>	文字列	<i>Expression</i> と <i>FiltExp</i> が、 <i>SampMode</i> が interpolated に設定されている場合に評価される頻度。値と時間単位を入力します。たとえば、時間間隔を 10m (10 分) とすると、10 分ごとの内挿値が取得されます。「 時間間隔の仕様 」を参照してください。
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、1 を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2、3、および 9 をサポートします(<i>Mode</i> が「minimum」、「maximum」、または「range」に設定されている場合)。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PIAdvCalcExpFilVal("sinusoid"+'sinusoidu',"y","t",'cdm158'="Manual","","total","time-weighted","compressed","10m",50,1,4,"")
```

PI ポイント cdm158 が昨日の午前 0 時から今日の午前 0 時までの値を使用して Manual に設定されているとき、デフォルトの PI Data Archive Server に存在する値を計算した式 'sinusoid'+'sinusoidu' の時間加重積算値を計算して返します。

関数は結果を 1 で乗算します。サンプリングモードが compressed であるため、関数はサンプリング間隔を無視し、sinusoid のヒストリイベントと sinusoid のヒストリイベントを併用して、フィルター式をサンプリングします。有効データの割合が 50% 以上である場合にのみ、関数は結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。

次の

[演算データ関数](#)

[フィルター式](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PIAdvCalcDat()

指定期間の PI ポイント値または PI AF 属性値を計算し、定間隔の値を返します。

構文

```
PIAdvCalcDat(DataItem, STime, ETime, Interval, Mode, CalcBasis, MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が計算されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、15m(15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「 時間間隔の仕様 」を参照してください。

引数	タイプ	説明
<i>Mode</i>	文字列	関数が実行する計算の種類: <ul style="list-style-type: none">• <code>total</code>• <code>minimum</code>• <code>maximum</code>• <code>stdev</code>• <code>range</code>• <code>count</code>• <code>average (time-weighted)</code>• <code>average (event-weighted)</code>
<i>CalcBasis</i>	文字列	計算方法: <ul style="list-style-type: none">• <code>time-weighted</code>• <code>event-weighted</code>
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、 <code>1</code> を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2、3、7、8、および 9 をサポートします(<i>Mode</i> が「 <code>minimum</code> 」、「 <code>maximum</code> 」、または「 <code>range</code> 」に設定されている場合)。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PIAdvCalcDat("sinusoid","y","t","3h","total","time-weighted",50, 1,4,"")
```

既定の PI Data Archive server にある値を計算して、PI ポイント `sinusoid` の時間加重積算値を返します。計算は、昨日の午前 0 時から今日の午前 0 時までの期間で、3 時間間隔で行われます。

関数は結果を 1 で乗算し、有効データの割合が 50% 以上である場合にのみ、結果を返します。計算した各積算値の右側に有効データ割合が表示されます。

次の

[演算データ関数](#)

[関数の手動定義](#)

引数の手動入力に関するガイドライン

PIAdvCalcFilDat()

指定期間の PI ポイント値または PI AF 属性値のフィルタリングと計算を行い、定間隔の値を返します。

構文

```
PIAdvCalcFilDat(DataItem, STime, ETime, Interval, FiltExp, Mode, CalcBasis, SampMode,  
SampFreq, MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を返す先の PI ポイント名または PI AF 属性名です。PI ポイント名または PI AF 属性名を含むセル範囲を指定すると、それぞれに対する値が計算されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、15m(15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用するブール型のパフォーマンスイクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。
<i>Mode</i>	文字列	関数が実行する計算の種類:
		<ul style="list-style-type: none">• <code>total</code>• <code>minimum</code>• <code>maximum</code>• <code>stdev</code>• <code>range</code>• <code>count</code>• <code>average (time-weighted)</code>• <code>average (event-weighted)</code>
<i>CalcBasis</i>	文字列	計算方法:
		<ul style="list-style-type: none">• <code>time-weighted</code>• <code>event-weighted</code>

引数	タイプ	説明
<i>SampMode</i>	文字列	<i>FiltExp</i> を評価するタイミングを決定するために使用されるサンプリング方法: <ul style="list-style-type: none">• compressed• interpolated
<i>SampFreq</i>	文字列	<i>FiltExp</i> が、 <i>SampMode</i> が interpolated に設定されている場合に評価される頻度。値と時間単位を入力します。たとえば、時間間隔を 10m (10 分) とすると、10 分ごとの内挿値が取得されます。 時間間隔の仕様 を参照してください。
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、 1 を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2、3、7、8、および 9 をサポートします(<i>Mode</i> が「 minimum 」、「 maximum 」、または「 range 」に設定されている場合)。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。

例

```
=PIAdvCalcFilDat("sinusoid", "y", "t", "3h", "'cdm158'=""Manual""", "total", "time-weighted", "compressed", "10m", 50, 1, 4, "")
```

PI ポイント **sinusoid** の時間加重積算値を、既定の PI Data Archive Server にある値を計算して返します。計算は、昨日の午前 0 時から今日の午前 0 時までの期間に 3 時間間隔で、PI ポイント **cdm158** が「**Manual**」に設定されているときに行われます。

関数は結果を 1 で乗算します。サンプリングモードが **compressed** であるため、関数はサンプリング間隔を無視し、**sinusoid** のヒストリイベントを使用して、フィルター式をサンプリングします。有効データの割合が 50%以上である場合にのみ、関数は結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。

次の

[演算データ関数](#)

[フィルター式](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

内挿値

PIAdvCalcExpDat()

指定期間に対してパフォーマンスイクエージョンに基づく計算を行い、定間隔の値を返します。

構文

```
PIAdvCalcExpDat(Expression, STime, ETime, Interval, Mode, CalcBasis, SampMode, SampFreq,  
MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンスイクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、15m(15分)と入力すると、対象期間中、15分の間隔で1つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>Mode</i>	文字列	関数が実行する計算の種類: <ul style="list-style-type: none">totalminimummaximumstdevrangecountaverage (time-weighted)average (event-weighted)
<i>CalcBasis</i>	文字列	計算方法: <ul style="list-style-type: none">time-weightedevent-weighted

引数	タイプ	説明
<i>SampMode</i>	文字列	<i>Expression</i> を評価するタイミングを決定するために使用されるサンプリング方法: <ul style="list-style-type: none">• compressed• interpolated
<i>SampFreq</i>	文字列	<i>Expression</i> が、 <i>SampMode</i> が interpolated に設定されている場合に評価される頻度。値と時間単位を入力します。たとえば、時間間隔を 10m (10 分) とすると、10 分ごとの内挿値が取得されます。 時間間隔の仕様 を参照してください。
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、 1 を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2、3、7、8、および 9 をサポートします(<i>Mode</i> が「 minimum 」、「 maximum 」、または「 range 」に設定されている場合)。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PIAdvCalcExpDat("'sinusoid'+'cdt158'","y","t","3h","total","time-weighted","compressed","10m",50,1,4,"")
```

昨日の午前 0 時から今日の午前 0 時の間の、既定の PI Data Archive Server に記録値が格納されているタイムスタンプに対して式 '**sinusoid**'+'**cdt158**' を計算し、さらにその計算値の時間加重積算値を 3 時間間隔で計算します。

関数は計算結果を 1 で乗算し、3 時間間隔の有効データの割合が 50% 以上であれば結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。サンプリングモードとして「**compressed**」が指定されているため、関数はサンプリング間隔を無視し、**sinusoid** または **cdt158** が記録値を格納している場合は常に値をサンプリングします。

次の

[演算データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

[内挿値](#)

PIAdvCalcExpFilDat()

指定期間に対してフィルタリングと式に基づく計算を行い、定間隔の値を返します。

構文

```
PIAdvCalcExpFilDat(Expression, STime, ETime, Interval, FiltExp, Mode, CalcBasis, SampMode, SampFreq, MinPctGood, CFactor, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が値を計算するパフォーマンス イクエージョン。「 式 」を参照してください。 複数の式を指定するには、完全な式が含まれているセルの範囲を入力します。この関数では、それぞれに値が返されます。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、15m(15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>FiltExp</i>	文字列	関数が値をフィルターするときに使用するブール型のパフォーマンス イクエージョンです。式が <code>false</code> と判定されると、関数は対応する値を除外します。「 フィルター式 」を参照してください。
<i>Mode</i>	文字列	関数が実行する計算の種類: <ul style="list-style-type: none">totalminimummaximumstdevrangecountaverage (time-weighted)average (event-weighted)
<i>CalcBasis</i>	文字列	計算方法: <ul style="list-style-type: none">time-weightedevent-weighted

引数	タイプ	説明
<i>SampMode</i>	文字列	<i>Expression</i> と <i>FiltExp</i> 評価するタイミングを決定するために使用されるサンプリング方法: <ul style="list-style-type: none">• compressed• interpolated
<i>SampFreq</i>	文字列	<i>Expression</i> と <i>FiltExp</i> が、 <i>SampMode</i> が interpolated に設定されている場合に評価される頻度。値と時間単位を入力します。たとえば、時間間隔を 10m (10 分) とすると、10 分ごとの内挿値が取得されます。「 時間間隔の仕様 」を参照してください。
<i>MinPctGood</i>	数値	値を計算して返すために必要とされる、対象期間中の有効データ割合の最小値。
<i>CFactor</i>	数値	関数が取得値に適用する係数。コンバージョンファクターが不要の場合は、 1 を入力します。時間加重の積算値を計算する場合は、記録周期を PI Server の既定時間単位(1 日単位)に変換する係数を指定します。
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2、3、7、8、および 9 をサポートします(<i>Mode</i> が「 minimum 」、「 maximum 」、または「 range 」に設定されている場合)。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PIAdvCalcExpFilDat("'sinusoid'+'cdt158'","y","t","3h","'cdm158'=""Manual""","total","time-weighted","compressed","10m",50,1,4,"")
```

既定の PI Data Archive Server から、昨日の午前 0 時から今日の午前 0 時までの 3 時間間隔で、式 'sinusoid'+'cdt158' の時間加重積算値を計算します。ただし、'cdm158' = "Manual" のときのみ計算対象とします。

関数は計算結果を 1 で乗算し、3 時間間隔の有効データの割合が 50% 以上であれば結果を返します。また、計算した積算値の右側に、有効データ割合を表示します。サンプリングモードとして「**compressed**」が指定されているため、関数はサンプリング間隔を無視し、**sinusoid** または **cdt158** が記録値を格納している場合、または **cdt158** の値が **Manual** に変化するか、この値から変化する場合は常に値をサンプリングします。

次の

[演算データ関数](#)

[フィルター式](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)[内挿値](#)

PITimeFilter()

指定期間内で式を定間隔で評価し、`true` と評価された合計時間を返します。

構文

```
PITimeFilter(Expression, STime, ETime, Interval, TimeUnit, OutCode, RootPath)
```

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が評価するブール型のパフォーマンス イクエージョンです。「 式 」を参照してください。 式を含むセル範囲を参照し、各式に対して値を取得します。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>Interval</i>	文字列	取得値間の間隔。間隔の長さを指定する値と時間単位を入力します。たとえば、 <code>15m</code> (15 分)と入力すると、対象期間中、15 分の間隔で 1 つの値が返されます。「 時間間隔の仕様 」を参照してください。
<i>TimeUnit</i>	文字列	計算結果の時間単位: <ul style="list-style-type: none"><code>seconds</code><code>minutes</code><code>hours</code><code>days</code>
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 3、2、7、および 8 に対応しています。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

```
=PITimeFilter($A$1,"y","t","1h","seconds",65,"thevax")
```

昨日の午前 0 時から今日の午前 0 時の期間の 1 時間間隔で、セル A1 の式が `true` と評価された秒数を、PI Data Archive Server (名前 thevax) から取得して、返します。

各計算間隔の開始時刻のタイムスタンプが、値の左側に表示されます。

次の

[条件指定時間データ関数](#)

[関数の手動定義](#)

[引数の手動入力に関するガイドライン](#)

PITimeFilterVal()

指定された期間内でパフォーマンス イクエージョンが `true` と評価された時間の合計を返します。

構文

`PITimeFilterVal(Expression, STime, ETime, TimeUnit, OutCode, RootPath)`

引数

引数	タイプ	説明
<i>Expression</i>	文字列	関数が評価するブール型のパフォーマンス イクエージョンです。「 式 」を参照してください。 式を含むセル範囲を参照し、各式に対して値を取得します。
<i>STime</i>	文字列	関数が値を計算する期間の開始時刻。「 時間入力 」を参照してください。
<i>ETime</i>	文字列	関数が値を計算する期間の終了時刻。「 時間入力 」を参照してください。
<i>TimeUnit</i>	文字列	計算結果の時間単位: <ul style="list-style-type: none">secondsminuteshoursdays
<i>OutCode</i>	整数	表示コードは、関数がどの追加データを返すか、およびどの向きに出力するかを決定します。コードを計算する方法については、「 表示コード 」を参照してください。 この関数は、ビット 2 とビット 3 に対応しています。
<i>RootPath</i>	文字列	The path to the data item referenced by the expression. See データアイテム .

例

`=PITimeFilterVal(A1,"y","t","seconds",0,"thevax")`

昨日の午前 0 時から今日の午前 0 時の期間でセル A1 の式が `true` と評価された秒数を、PI Data Archive Server (名前 thevax) から取得して、返します。

次の

[条件指定時間データ関数](#)

[関数の手動定義](#)[引数の手動入力に関するガイドライン](#)

入力関数

PI Data Archive または PI AF に値を書き込む関数を入力します。

PIPutVal() や PIPutValX()

PI Data Archive Server または PI AF Server に値を書き込み、ステータスを示す文字列を返します。

- Visual Basic for Applications (VBA) 内では PIPutVal 関数と PIPutValX 関数のみ使用します。Excel の数式バーにこれらの関数を入力しないでください。「[データ書き出し用の関数](#)」を参照してください。
- 新バージョンの PI DataLink では、PIPutVal 関数と PIPutValX 関数に違いはありません。PI DataLink の、PI DataLink 2013 より前のバージョンでは、PIPutVal 関数は、実数、整数、デジタルの 3 つの PI ポイントのタイプのみをサポートしています。これらのバージョンの PI DataLink では、PIPutValX 関数が、文字列のポイントとミリ秒単位のタイムスタンプの表示をサポートするために追加されています。

構文

```
PIPutVal(DataItem, Value, TimeStamp, RootPath, OutCell)  
PIPutValX(DataItem, Value, TimeStamp, RootPath, OutCell)
```

引数

引数	タイプ	説明
<i>DataItem</i>	文字列	関数が値を書き込む PI ポイント名または PI AF 属性名です。
<i>Value</i>	参照	書き込む値を含むセルへの参照です。文字列または数値で指定します。
<i>TimeStamp</i>	文字列	値に対するタイムスタンプ。「 時間入力 」を参照してください。
<i>RootPath</i>	文字列	データアイテムへのパス。「 データアイテム 」を参照してください。
<i>OutCell</i>	参照	マクロ関数が取得した文字列を表示するセルの参照。様々な文字列が返されます。 <ul style="list-style-type: none">書き込みが成功した場合:<ul style="list-style-type: none">PIPutVal 関数は、<code>xxx Written</code> を返します。<code>xxx</code> には、値のタイプに応じて、「Real Value」、「Integer」、「Digital State」、または「Value」が設定されます。PIPutValX 関数は、<code>Value Written</code> を返します。書き込みが失敗した場合、関数は、<code>[-5] Tag not found</code> のようなエラーコードを返します。

注意

`PIPutVal` 関数と `PIPutValX` 関数は、設定属性である PI AF 属性には書き込みできません。上記の関数は、**[Configuration Item]** チェックボックスがオフになっている PI AF 属性にのみ書き込みできます。PI System Explorer の**[属性]**タブでチェックボックスの状態を確認できます。

`PIPutVal` 関数と `PIPutValX` 関数は、PI Data Archive のすべてのデータタイプに対応しています。タイムスタンプは、クライアントコンピューターの PI Data Archive 用のタイムゾーン情報を使用して変換されます。ただし、**[PI Data Archive のタイムゾーン]** 設定が有効である場合を除きます。

`PIPutVal()` は、書き込む前に、タグのユーザー指定範囲内で値が書かれているかは確認しません。値の範囲は、`zero` から `zero + span` として定義され、`zero` と `span` は、ポイント作成時に指定されるポイント属性です。関数は `Value Written` 文字列を返します。`Over Range` または `Under Range` が PI Data Archive に格納されている場合でも同じです。

廃止された関数

一部の関数が、より機能性の高い関数に代わりました。廃止された関数を使用している従来のワークシートを引き続き使用できるよう、PI DataLink では従来の廃止された関数もサポートしていますが、OSIsoft では対応する新しい関数を使用することをお勧めします。

従来の関数	新しい関数
<code>PICalcVal</code>	<code>PIAdvCalcVal</code>
<code>PICalcDat</code>	<code>PIAdvCalcDat</code>
<code>PIExTimeVal</code>	<code>PIArcVal</code> (<code>Mode</code> が「 <code>exact time</code> 」に設定されている場合)

章 9

トラブルシューティング

PI DataLink が正しく動作しない場合は、以下の項を参照してください。問題に対応する項目がない場合や、問題を解決できない場合は、テクニカルサポートまでご連絡ください([OSIsoft Customer Portal](#) を参照)。

ログファイル

PI DataLink は、一般的なログファイルを生成しません。ワークシートのエラーメッセージを使用して、問題を診断できます。ただし、ログファイルを生成するプロセスもあります:

- インストール

`../pipc/dat` ディレクトリには、インストールログファイル、**SetupPIDataLink_x64.log**、および **SetupPIDataLink_x86.log** が含まれます。

- PI Data Archive Server 接続

PI SMT の PI Data Archive ログファイルで、PI Data Archive Server 接続における問題について検証してください。

PI DataLink タブが使用できない

PI DataLink をインストールしても **[PI DataLink]** タブが表示されない場合は、Excel のアドインを手動で設定([Microsoft Excel アドイン設定](#)を参照)するか、アドインのセキュリティ設定を下げる必要があります([セキュリティ](#)を参照)。

Excel ワークシート(またはアドイン)のユーザー作成 VBA スクリプトの中には、Excel リボンをリセットするものがあります。この場合、PI DataLink アドインがロードされていても、**[PI DataLink]** タブが表示されないことがあります。**[PI DataLink]** タブ自体が使用できなくても、埋め込まれた PI DataLink の配列関数は機能します。通常、Excel を再起動すると、この問題は解決します。

配列とセルの制限

PI DataLink では、計算配列中の要素数は Excel の制限に従います。1 つの PI DataLink 関数が 1 回の検索で返すことができる値の数は、1,048,576 個以下です。

PI DataLink が使用する Excel SDK は、文字列の長さを 255 文字までに制限しています。参照セルに PI DataLink 式を入力する場合、255 文字の制限を超えないようにしてください。

同様に、PI DataLink は、出力される文字列(PI ポイント名)を 255 文字で切り捨てます。

データ取得制限

エラーコード-11091 は、PI DataLink 関数によって実行されたクエリが、PI Data Archive から *ArcMaxCollect* チューニングパラメーターで許可されている記録値よりも多く取得しようとしたことを示しています。

この制限は、PI Data Archive から取得した記録された値に適用されるので、少數の値を返す関数であっても、このチューニングパラメーターによる制約を超過してしまうことがあります。たとえば、PI ポイント *sinusoid* の値が、0 から 100 の間で定間隔に変化するとします。1 年間、30 秒ごとに、*sinusoid* の定間隔ヒストリデータを、95 未満の値を除外するフィルター式付きで取得する内容のクエリを作成できます。このクエリによって取得される値の個数は少ないかもしれません、クエリの処理上、それ以上に多数の記録されている値が取得されていることになります。

このエラーが表示された場合、検索条件を修正して、取得データを減らすことを検討してください。たとえば、関数がデータを取得する期間を短くしてみてください。

関数でサポートされないデータアイテム

メッセージ `Data item not supported by function` が表示された場合、以下のことを示しています:

- PI AF 属性が、計算を許容していないカスタムデータ参照を使用しています。PI DataLink で使用するには、カスタムデータ参照コードを変更して、データアクセス方法の使用を許可してください。これは既定では無効です。
- 演算データ関数は、PI ポイント配列タイプの PI AF 属性でサマリー計算を試みました。

重複イベントの存在

メッセージ `Duplicate event exists: specify index for EventName` は、このワークシートの行の親イベントに `EventName` という名前の子イベントが複数含まれていることを示しています。これを解決するには、イベントの比較関数の指定を変更し、この列に表示されている属性にインデックス付きバージョンのイベントを使用します。

親イベントに重複した名前を持つ子イベントがある場合、PI DataLink は、[属性の追加] ウィンドウにある子イベント名にインデックスを自動的に追加します。同じレベルで他のイベントに重複した名前がない場合、それらの名前にはインデックスは付きません。詳細は、「[子イベント属性を列として\[イベントの比較\]タスクペインに追加](#)」を参照してください。

セキュリティ

アドイン、ActiveX コントロールおよびマクロのための Excel セキュリティ機能は、Excel 内部で実行できるコンポーネントタイプを管理しています。セキュリティ設定では、「各オブジェクトを無効にする」、「メッセージを表示してから有効にする」、「完全に有効にする」のいずれかを設定します。

Excel セキュリティ設定は、PI DataLink 関数の特定の機能と矛盾することがあります。

- PI DataLink は Excel アドインです。セキュリティ設定でアドインの無効化が行われると、PI DataLink を Excel に読み込めません。
- PIPutVal() や PIPutValX() 関数を含むワークシートはマクロを使用するため、マクロを有効にする必要があります([PIPutVal\(\)](#) や [PIPutValX\(\)](#) を参照)。

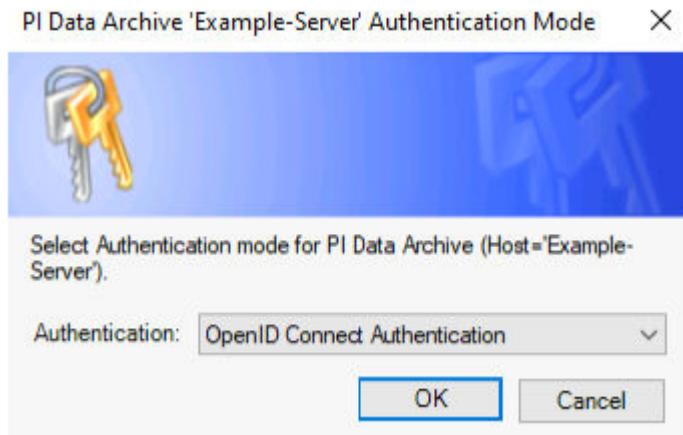
一般的に Excel セキュリティをワークシートに適応する場合や、PI DataLink 関数の使用で問題が発生した場合、PI DataLink の使用計画に沿って Excel セキュリティ設定を下げる必要があります。

Excel セキュリティの詳細については、Microsoft Excel オンラインヘルプまたはオンラインリソースを参照してください。

OIDC 認証との接続

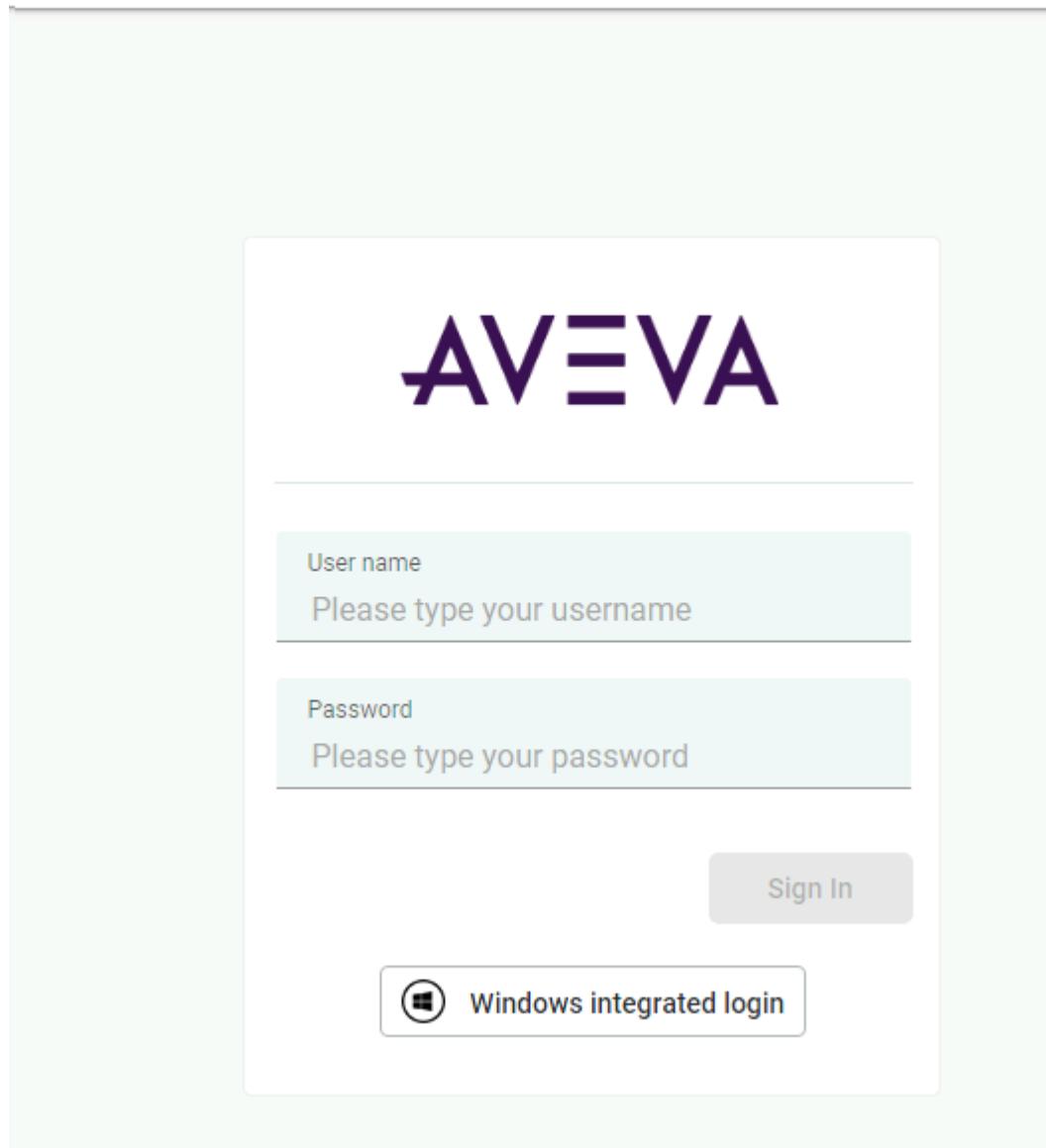
PI DataLink 2023 では、PI Server 2023 以降で Open ID Connect (OIDC) 認証プロトコルを使用して接続する機能が導入されました。OIDC を使用して接続するには、PI DataLink 2023 の PI Server でインストール後の追加設定が必要です。これらの設定手順については、「PI Server 2023 ユーザーガイド」を参照してください。

1. まず PI Server に接続して OIDC をサポートするように設定されている場合、認証ダイアログが表示されます。OpenID Connect 認証または Windows 認証のいずれかを選択できます。選択した認証モードは、そのサーバーへの次回の接続開始時に記憶されています。



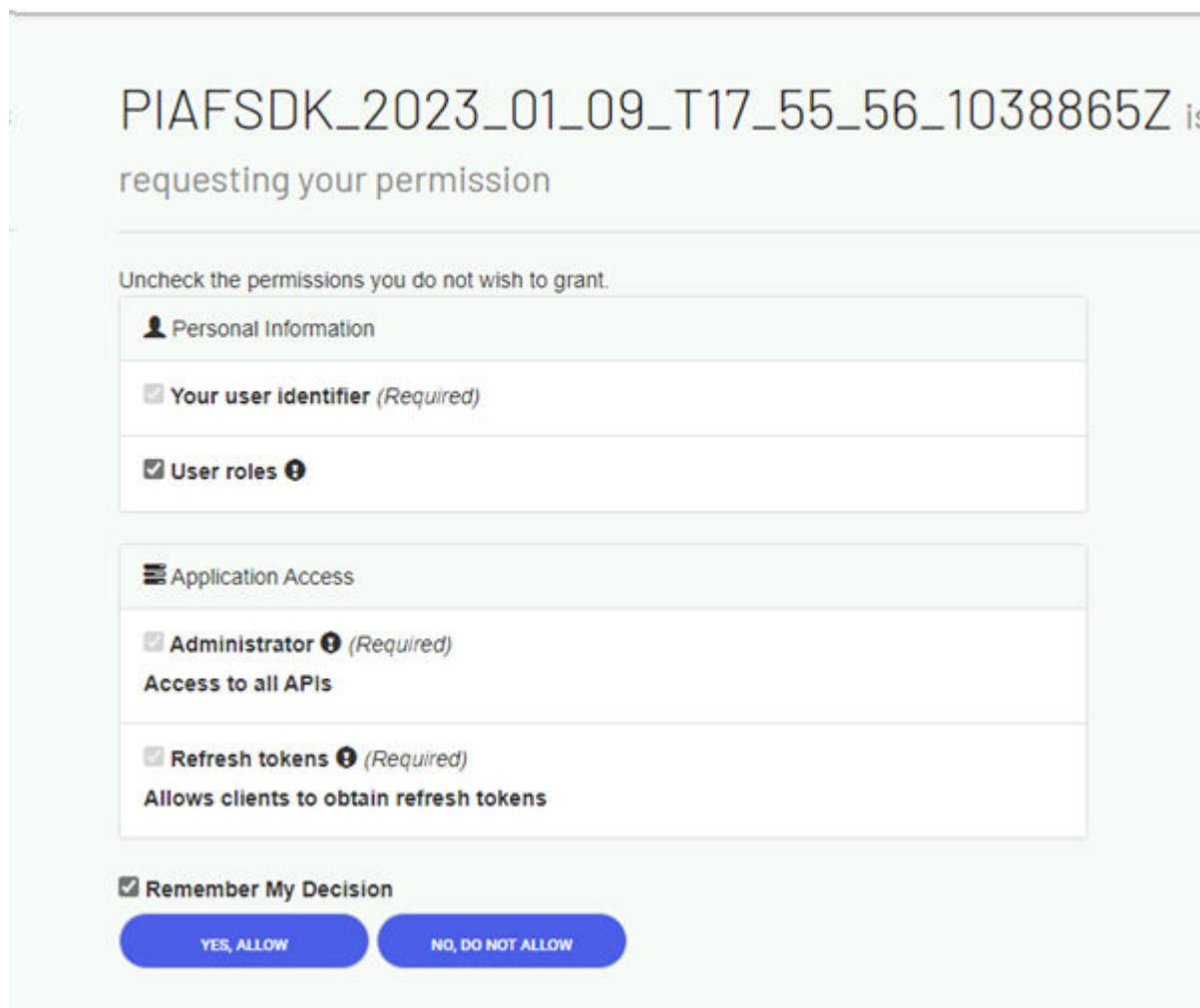
2. **OIDC 認証**を選択するとブラウザウィンドウが開き、資格情報でログインするよう求められます。

AVEVA™ Identity Manager



3. 資格情報を入力した後、資格情報の使用に同意するか尋ねられます。

AVEVA™ Identity Manager



4. OIDC 認証に使用されるトークンの有効期限は、設定期間後に切れます。使用期間が経過した後に次に PI DataLink を使用するとブラウザーウィンドウが再度開き、資格情報を使用して再度ログインする必要があります。

認証モードはいつでも **OIDC 認証** と **Windows 認証** の間で切り替えることができます。[設定] メニュー内のリンクから [Connection Manager] に移動できます。

5. **Connection Manager** を開いたら、[PI Data Archiver] または [AF Server] を右クリックして [**Connect As (接続)**] を選択します。次に認証モードダイアログが表示され、モードを切り替えることができます。**OIDC 認証モード**を選択した場合は、ログインプロンプトに従い、前述の指示に従う必要があります。

PI Data Archive の接続に関する問題

本バージョンの PI DataLink は、PI Data Archive へ接続する際、強力なセキュリティ設定を使用します。新しい設定では、PI パスワード認証に関するセキュリティ上の脆弱性に対応しています。詳しくは、

OSIsoft テクニカルサポートの警告「[AL00206 - Security Alert: PI Authentication Weakness](#)」(セキュリティ警告 : PI 認証の脆弱点) を参照してください。

PI DataLink が以下のものを使用すると、新セキュリティ設定ではエラーとなります:

- PI Data Archive に接続する PI ユーザー名とパスワード
- PI Data Archive に接続する既定のユーザー

特に、これらの設定で次のエラーが発生する可能性があります。

Cannot connect to the PI Data Archive. Windows authentication trial failed because insufficient privilege to access the PI Data Archive. Trust authentication trial failed because insufficient privilege to access the PI Data Archive.

このようなエラーを解決するには、次の操作を行います:

- PI Data Archive (PI Server 2023 以降) に接続するユーザーの Open Id Connect Mapping を設定します。PI Server 2023 以降で PI DataLink を使用する場合は、Open Id Connect 認証を使用することをお勧めします。OIDC 認証を使用するには、PI Server と AF Server で Open ID Connect Role PI Mappings を作成する必要があります。

PI DataLink を実行する各コンピューターで OIDC 認証プロトコルを有効にする必要があります。

- PI Data Archive に接続するユーザーの PI Mapping を設定します。

この解決法(PI Data Archive のバージョン 3.4.380 以降で使用可能)を使用することをお勧めします。これは最も安全な接続方法です。PI マッピングの作成について詳しくは、PI Server のトピック「マッピング管理」を参照してください。PI Mapping を使用するには、PI DataLink を実行する各コンピューターで Windows Security 認証を有効にする必要があります。

- PI Data Archive に接続するユーザーの PI Trust を設定します。

これは次善の策です。PI Trust を使用するには、PI DataLink を実行する各コンピューターで PI Trust 認証を有効にする必要があります。

- PI Mapping や PI Trust が使用できない場合は、明示的ログインプロンプトを許容するように各コンピューターを設定します。

この方法はお勧めできません。明示的ログインは安全ではないためです。この方法では、特定の Microsoft Excel セッションで最初に各 PI Data Archive に接続したときにユーザー名とパスワードを入力する必要があります。

詳しくは、PI Server のトピック「認証の管理」を参照してください。

注: ログインプロンプトで間違ったパスワードを入力しても、同じエラーメッセージが表示されます。

認証プロトコルの有効化

注: PI Server 2023 以降を使用する場合、このセクションは OIDC 認証ではなく Windows 認証を選択した場合にのみ適用されます。

定義済みの PI mapping や PI trust で PI Data Archive に接続するには、PI System Explorer を使用して、コンピューターで対応するプロトコルを有効にします。PI Mapping や PI Trust を使用して PI Data Archive に接続する各コンピューターにて設定する必要があります。

1. [スタート] > [PI System] > [PI System Explorer] をクリックして PI System Explorer を起動します。
2. [ツール] > [オプション] をクリックして [オプション] ダイアログを開きます。

3. [オプション]ダイアログの[サーバーオプション]タブを選択します。
4. [PI Data Archive 接続設定]セクションで、配列キーとチェックボックスを使用して[プロトコル]に一覧表示されているプロトコルをソートおよび有効化します。
 - PI mapping を使用するには、[Windows セキュリティ]がチェックされている必要があります。PI Mapping を[Protocol order]の最初のプロトコルにしてください。
 - PI trust を使用するには、[PI Trust]がチェックされている必要があります。
5. [OK]をクリックします。

明示的ログインプロンプトを許可するためのコンピューター設定

PI User としてログインして PI Data Archive に接続するには、PI System Explorer を使用して、使用しているコンピューターでの明示的ログインプロンプトを許可します。PI ユーザーとしてログインして PI Data Archive に接続する各コンピューターにて設定する必要があります。

注: より安全な PI Data Archive への接続方法を選択することを推奨します。

1. [スタート] > [PI System] > [PI System Explorer]をクリックして PI System Explorer を起動します。
2. [ツール] > [オプション]をクリックして[オプション]ダイアログを開きます。
3. [オプション]ダイアログの[サーバーオプション]タブを選択します。
4. [PI Data Archive 接続設定]セクションの[ログインプロンプトを許可]チェックボックスを選択します。
5. [OK]をクリックします。

ユーザー名とパスワードを入力して、Excel の新しいセッションを開始するたびに PI Data Archive にアクセスします。接続プロンプトには、検索ツールを使って PI Data Archive を検索するか、接続管理から手動でアクセスできます。「[接続管理からログイン資格情報を入力する](#)」を参照してください。

接続管理からログイン資格情報を入力する

PI ユーザーとしてログインし、PI Data Archive に接続した場合、各 Excel セッションで接続する PI Data Archive ごとに、ユーザー名とパスワードを入力する必要があります。

1. [明示的ログインプロンプトを許可するためのコンピューター設定](#)。
2. [PI DataLink]タブの[リソース]グループで[設定]をクリックして、[設定]ウィンドウを開きます。
3. [接続管理]をクリックして、[サーバー]ウィンドウを開きます。
4. サーバーのリストで、エラーを返した PI Data Archive Server を右クリックして[設定して接続]をクリックし、[PI Data Archive に接続]ウィンドウを開きます。
5. [認証]リストから、[PI ユーザー認証]を選択します。

ウィンドウに[認証]リストが表示されない場合は、ログインプロンプトを表示するようにコンピューターを設定する必要があります。

6. ユーザー名とパスワードを入力して、[OK]をクリックします。

タイムアウト例外エラーの解決

大きなポイントデータベースを持つ PI Data Archive Server を検索すると、PI Data Archive がすべての値を取得する前にデータタイムアウトが発生することがあります。この場合、次のようなエラーが返されます。

`OSIsoft.PI.Net.PITimeoutException: [-10722] PINET: Timeout on PI RPC or System Call.`

エラーを解決するには、接続管理を使用して、データタイムアウト値を増やします。

1. [PI DataLink]タブの[リソース]グループで[設定]をクリックして、[設定]ウィンドウを開きます。
2. [接続管理]をクリックして、[サーバー]ウィンドウを開きます。
3. エラーを返した PI Data Archive Server を右クリックして[プロパティ]をクリックし、[PI Data Archive プロパティ]ウィンドウを開きます。
4. [データ タイムアウト]フィールドの値を増やして、[OK]をクリックします。

更新されないデータ

PI DataLink は PI ポイントの構成データ、および PI AF の非時間ベースのデータ(PI ポイントデータ参照およびイベント以外のデータ)をキャッシュします。これにより、ワークシートに挿入したデータが、関数を再計算しても、変更された設定または非時間ベースのデータに基づいた値を表示するよう更新されない場合があります。関数を計算する前に、PI DataLink で、キャッシュが最後に消去された時刻をチェックします。キャッシュが過去 6 か時間以内に消去されていない場合は、自動的にキャッシュが消去されます。キャッシュをクリアして、更新後のデータを素早く取得するには、次のいずれかを行います。

- [Clear Cache (キャッシュをクリア)] ([設定] ウィンドウ)をクリックします。「[Excel の PI DataLink 設定を管理する](#)」を参照してください。
- Microsoft Excel を閉じ、再起動します。