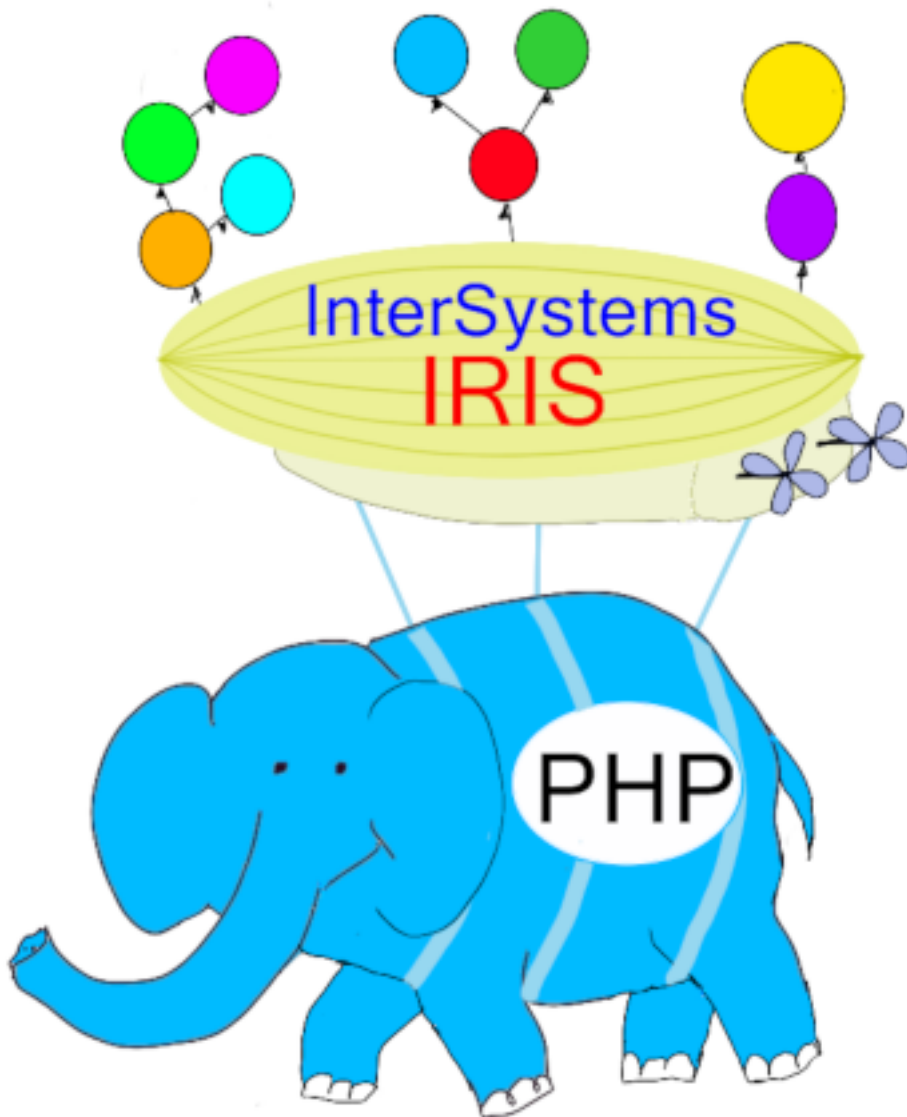


記事

[Toshihiko Minamoto](#) · 2021年8月18日 13m read

[Open Exchange](#)

## InterSystems IRISで階層型データを使用するためのPHPモジュール



PHPはその公開当初から、多くのライブラリや市場に出回っているほぼすべてのデータベースとの統合をサポートしていることでよく知られています(またそのことで批判を受けてもいます)。にもかかわらず、何らかの不可解な理由により、グローバル変数については階層型データベースをサポートしませんでした。

グローバル変数は階層情報を格納するための構造です。Key-Value型データベースにある程度似ていますが、キーを次のようにマルチレベルにできるという点で異なります。

```
Set ^inn("1234567890", "city") = "Moscow"  
Set ^inn("1234567890", "city", "street") = "Req Square"  
Set ^inn("1234567890", "city", "street", "house") = 1  
Set ^inn("1234567890", "year") = 1970  
Set ^inn("1234567890", "name", "first") = "Vladimir"
```

```
Set ^inn("1234567890", "name", "last") = "Ivanov"
```

この例では、

マルチレベルの情報は、ビルトインのObjectScript言語を使用して、グローバル変数の^innに保存されています。グローバル変数^innは、ハードドライブに保存されています（これは、最初の「^」記号で示されています）。

PHP からグローバル変数进行操作するには、PHP

モジュールによって追加される新しい関数が必要となります。これについて、以下で説明します。

グローバル変数は、階層を操作するための多数の関数をサポートしています。固定レベルと縦型のトラバーサルツリーでツリー全体と個別のノードの削除、コピー、および貼り付けを行えます。

また、質の高いデータベースと同様に、ACID トランザクションもサポートされています。

これらは次の2つの理由に

より、非常に迅速に行われます（一般的なPCで、1秒間に $10^5 \sim 10^6$ の挿入が行われます）。

1. グローバル変数は SQL に比べると、より低レベルの抽象化である。
2. ベースは数十年の間グローバルスコープで稼働しており、この間に洗練され、コードは完全に最適化されてきた。

グローバル変数について詳しくは、「[グローバル変数: データ管理の魔法の剣](#)」という連載記事をご覧ください。

[パート1.](#)

[ツリー。パート2.](#)

[スパスアレイ。パート3.](#)

この業界では、グローバル変数は主に、医療、個人データ、銀行などの構造化されていないまばらな情報のストレージシステムで使用されています。

私はPHPを気に入っており、開発作業でも使用しているため、グローバル変数を使って色々と試してみたいと思いました。IRISとCaché用のPHPモジュールは存在しなかったため、InterSystemsに問い合わせ、作成するよう依頼しました。InterSystemsは教育助成金の一環として開発を後援してくれたおかげで、私の院生とともにモジュールを作成することになりました。

一般的に、InterSystems IRISはマルチモデルDBMSであるため、ODBCを通じてSQLを使ってPHPから操作することができますが、私が興味を持っていたのはグローバルであったため、それに使用できるコネクタは存在しなかったのです。

それはさておき、このモジュールはPHP 7.xで利用できます（7.0~7.2でテストしました）。

現在、同じホストにインストールされているInterSystems IRISとCachéでのみ動作します。

[OpenExchangeのModuleページ](#)（InterSystems

IRISとCachéの開発者向けのプロジェクトとアドオンのディレクトリ）。

開発者同士で関連する体験をシェアできるDISCUSSセクションが設けられています。

こちらからダウンロードしてください。

<https://github.com/intersystems-community/phpextiris>

コマンドラインからリポジトリをダウンロードする場合:

```
git clone https://github.com/intersystems-community/php_ext_iris
```

[モジュールのインストール手順（英語・ロシア語）。](#)

モジュールの関数:

PHP関数	説明
-------	----

データの操作	
irisset(\$node, value)	<p>ノードの値を設定します。irisset(\$global, \$subscript1, ..., \$subscriptN, \$value); irisset(\$global, \$value); 戻り値: true または false (エラーの場合)。この関数のすべてのパラメーターは文字列か数値です。最初のパラメーターはグローバルの名前で、次にインデックス、そして最後のパラメーターは値です。irisset('^time',1); irisset('^time', 'tree', 1, 1, 'value'); ObjectScript equivalent: Set ^time = 1 Set ^time("tree", 1, 1) = "value" irisset(\$arrayGlobal, \$value);パラメーターは2つしかありません。1つ目はグローバルの名前とすべてのインデックスを含む配列で、2つ目は値です。\$node = ['^time', 'tree', 1, 1]; irisset(\$node,'value');</p>
irisget(\$node)	<p>ノードの値を取得します。戻り値: 値 (数値または行)、NULL (値が定義されていません)、またはFALSE (エラーの場合)。irisget(\$global, \$subscript1, ..., \$subscriptN); irisget(\$global); この関数のすべてのパラメーターは行または数値です。1つ目はグローバルの名前で、残りはサブスクリプトです。グローバルにはサブスクリプトがない場合があります。\$res = irisget('^time'); \$res1 = irisget('^time', 'tree', 1, 1); irisget(\$arrayGlobal); 唯一のパラメーターは、グローバルの名前とそのすべてのサブスクリプトが格納されている配列です。\$node = ['^time', 'tree', 1, 1]; \$res = irisget(\$node);</p>
iriszkill(\$node)	<p>ノードの値を削除します。戻り値: TRUEまたはFALSE (エラーの場合)。この関数はノードの値のみを削除し、下位のブランチには影響しないことに注意してください。iriszkill(\$global, \$subscript1, ..., \$subscriptN); iriszkill(\$global); この関数のすべてのパラメーターは行または数値です。1つ目はグローバルの名前で、残りはサブスクリプトです。グローバルにはサブスクリプトがない場合があります。\$res = iriszkill('^time'); // 下位ブランチは削除されません。\$res1 = iriszkill('^time', 'tree', 1, 1); iriszkill(\$arrayGlobal); 唯一のパラメーターは、グローバルの名前とそのすべてのサブスクリプトが格納されている配列です。\$a = ['^time', 'tree', 1, 1]; \$res = iriszkill(\$a);</p>
iriskill(\$node)	<p>ノードとすべての子孫ブランチを削除します。戻り値: TRUEまたはFALSE (エラーの場合)。iriskill(\$global, \$subscript1, ..., \$subscriptN); iriskill(\$global); この関数のすべてのパラメーターは行または数値です。1つ目はグローバルの名前で、残りはインデックスです。グローバルにはインデックスがない場合があり、その場合は完全に削除されます。\$res1 = iriskill('^example', 'subscript1', 'subscript2'); \$res = iriskill('^time'); // グローバルは完全に削除されます。iriskill(\$arrayGlobal); 唯一のパラメーターは、グローバルの名前とそのすべてのサブスクリプトが格納されている配列です。\$a = ['^time', 'tree', 1, 1]; \$res = iriskill(\$a);</p>
irisorder(\$node)	<p>指定されたレベルでグローバルのブランチをトランプスします。戻り値: 同じレベルのグローバルの前のノードのフルネームが格納されている配列、またはFALSE (エラーの場合)。irisorder(\$global, \$subscript1, ..., \$subscriptN); この関数のすべてのパラメーターは文字列または数値です。1つ目のパラメーターはグローバルの</p>

	<p>名前、残りはサブスクリプトです。PHPとObjectScript相当の使用法: <code>irisorder('^ccc','new2','res2');</code> // <code>\$Order('^ccc("new2", "res2")) irisorder(\$arrayGlobal);</code> 唯一のパラメータは、グローバルの名前と最初のノードのサブスクリプトが格納されている配列です。 <code>\$node = ['^inn', '1234567890', 'city']; for (; \$node !== NULL; \$node = irisorder(\$node)) { echo join(', ', \$node).' '.irisget(\$node)." "; }</code> Returns: <code>^inn, 1234567890, city=Moscow ^inn, 1234567890, year=1970</code></p>
<code>irisorderrev(\$node)</code>	<p>指定されたレベルでグローバルのブランチを逆順にトラバースします。戻り値: 同じレベルのグローバルの前のノードのフルネームが格納されている配列、またはFALSE (エラーの場合)。 <code>irisorderrev(\$global, \$subscript1, ..., \$subscriptN);</code> この関数のすべてのパラメータは行または数値です。1つ目はグローバルの名前で、残りはサブスクリプトです。PHPとObjectScript相当の使用法: <code>irisorderrev('^ccc','new2','res2');</code> // <code>\$Order('^ccc("new2", "res2"), -1) irisorderrev(\$arrayGlobal);</code> 唯一のパラメータは、グローバルの名前と最初のノードのサブスクリプトが格納されている配列です。 <code>\$node = ['^inn', '1234567890', 'name', 'last']; for (; \$node !== NULL; \$node = irisorderrev(\$node)) { echo join(', ', \$node).' '.irisget(\$node)." "; }</code> 戻り値: <code>^inn, 1234567890, name, last=Ivanov ^inn, 1234567890, name, first=Vladimir</code></p>
<code>irisquery(\$CmdLine)</code>	<p>グローバルの縦型トラバース 戻り値: 下位ノード (使用可能な場合) またはグローバルの次のノード (埋め込みノードがない場合) のフルネームが含まれている配列。 <code>irisquery(\$global, \$subscript1, ..., \$subscriptN);</code> この関数のすべてのパラメータは文字列または数値です。1つ目はグローバルの名前で、残りはサブスクリプトです。PHPとObjectScript相当の使用法: <code>irisquery('^ccc','new2','res2');</code> // <code>\$Query('^ccc("new2", "res2")) irisquery(\$arrayGlobal);</code> 唯一のパラメータは、グローバルの名前と最初のノードのインデックスが格納されている配列です。 <code>\$node = ['^inn', 'city']; for (; \$node !== NULL; \$node = irisquery(\$node)) { echo join(', ', \$node).' '.irisget(\$node)." "; }</code> 戻り値: <code>^inn, 1234567890, city=Moscow ^inn, 1234567890, city, street=Req Square ^inn, 1234567890, city, street, house=1 ^inn, 1234567890, name, first=Vladimir ^inn, 1234567890, name, last=Ivanov ^inn, 1234567890, year=1970</code> この順序は挿入時にグローバル内で自動的に昇順にソートされるため、設定した順序とは異なります。</p>
<b>サービス関数</b>	
<code>irissetdir(\$FullPath)</code>	<p>データベースのディレクトリをセットアップします。戻り値: TRUEまたはFALSE (エラーの場合)。 <code>irissetdir('/InterSystems/Cache/mgr');</code> これはデータベースに接続する前に実行する必要があります。</p>
<code>irisexec(\$CmdLine)</code>	<p>データベースコマンドを実行します。戻り値: TRUEまたはFALSE (エラーの場合)。 <code>irisexec('kill ^global(6));</code> // グローバルを削除するObjectScriptコマンド</p>
<code>irisconnect(\$login, \$pass)</code>	データベースに接続します。
<code>irisquit()</code>	DBとの接続を閉じます。
<code>iriserrno()</code>	エラーコードを取得します。
<code>iriserror()</code>	エラーのテキストによる説明を取得します。

モジュールを試してみたい場合は、dockerコンテナの実装などを確認してください。

特にDCや使用したい方のために、Caché用php-moduleがセットアップされた仮想マシンを実行しています。

InterSystems Cachéにモジュールを自分でインストールする場合

単なる好奇心で、私のPC (AMD FX-9370@4700Mhz 32GB、LVM、SATA SSD) のdockerコンテナで新しい値をデータベースに挿入する速度をチェックするプリミティブテストを2つ実行しました。

- 100万個の新しいノードをグローバルに挿入するのに、1.81秒掛かりました (1秒あたり552Kの挿入)。
- 同じグローバルの値を1,000,000回更新するのに、1.98秒掛かりました (1秒あたり505Kの更新)。  
興味深かったのは、挿入が更新よりも早く行われたということです。  
どうやらこれは、迅速な挿入を目的としたデータベースの初期最適化の結果のようです。

明らかに、これらのテストは原始的であり、コンテナ内で実行されるため、100%の正確性または有用性があるとは考えられません。 PCIe SSDにディスクシステムを備えたより強力なハードウェアでは、1秒あたり数千万の挿入を達成可能です。

## 作成中の機能とその状況

1. トランザクションを操作するための便利な関数を追加できます (irisexecで使用できます)。
2. グローバル構造全体を返す関数は実装されていません。PHPからグローバルをトラバースする関数についても同様です。
3. PHP配列をサブツリーとして保存する関数は実装されていません。
4. ローカルデータベース変数へのアクセスは実装されていません。  
irisetを使用した方が良いですが、irisexecのみを使用してください。
5. 逆順での縦型グローバルトラバースは実装されていません。
6. メソッドを使ったオブジェクト経由のデータベースアクセス (現在の関数に類似) は実装されていません。

現在のモジュールはまだ本番対応とは言えません。高負荷やメモリリークについてのテストは行われていません。ただし、必要だという方がいらっしゃれば、いつでもご連絡ください (Sergey Kamenev宛: [sukamenev@gmail.com](mailto:sukamenev@gmail.com))。

## 結論

グローバルは特定のデータタイプ (医療、個人データなど) に強力で高速な機能を提供しているにも関わらず、長い間、PHPの世界とグローバル変数での階層型データベースの世界では、実質的に重なることはありませんでした。

このモジュールをきっかけに、PHPプログラマーがグローバル変数を試すようになり、ObjectScriptプログラマーがPHPでウェブインターフェースを簡単に開発できるようになることを願っています。

追伸 最後までお読みいただき、ありがとうございました！

[#API](#) [#Code Snippet](#) [#SOAP](#) [#キーバリュー](#) [#クラウド](#) [#Caché](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)

[#Open Exchange](#) [#その他](#)

[InterSystems Open Exchangeで関連アプリケーションを確認してください](#)

---

ソースURL:<https://jp.community.intersystems.com/post/intersystems-iris%E3%81%A7%E9%9A%8E%E5%B1%A4%E5%9E%8B%E3%83%87%E3%83%BC%E3%82%BF%E3%82%92%E4%BD%BF%E7%94%A8%E3%81%99%E3%82%8B%E3%81%9F%E3%82%81%E3%81%AEphp%E3%83%A2%E3%82%B8%E3%83%A5%E3%83%BC%E3%83%AB>