

# Legend

Riconciliazione bancaria e gestione incassi

Versione 1.00

di Rodolfo Calzetti (postmaster@rudyz.net)

Analizziamo due problematiche aziendali che possono essere affrontate con lo stesso strumento: il clustering dei movimenti. Legend è un modulo di Corsaro integrato con la gestione Pi Cubo (vedi documento relativo), che consente di individuare e inserire in una pratica tutti i movimenti che ineriscono ad una stessa situazione. Faremo due esempi.

## **Riconciliazione bancaria**

Si configurano due liste; una verrà popolata dai movimenti come sono registrati dall'azienda: l'altra dai movimenti ricevuti da remote banking.

Il software consente di abbinare le due immagini dello stesso movimento e controllare, così, l'operato della banca.

Le liste, in tale configurazione, sono due e di segno opposto in modo da quadrare a zero. In pochissimi istanti il programma è in grado di abbinare migliaia di movimenti.

## **Gestione incassi**

Questa applicazione dei processi di clustering è più interessante, specialmente per quelle aziende che offrono servizi al dettaglio.

Da una parte ci sono le fatture (o i tributi, se trattasi di Pubblica Amministrazione) che esprimono un "valore" che l'azienda dà al cliente/utente sotto forma di prodotto o di servizio.

Dall'altra ci sono i pagamenti del cliente, effettuati in tempi successivi all'emissione della fattura/cartella.

Gli incassi potrebbero essere complessi. Un bonifico potrebbe pagare molte fatture; il file CBI proveniente dalla banca potrebbe non aver conservato il riferimento ai numeri fattura. In tal caso non sarebbe possibile un riconoscimento uno a uno. Il programma è però dotato di un algoritmo esclusivo che è in grado di ovviare alla complessità del cosiddetto "problema delle somme parziali" che consiste nell'estrarre da un insieme di numeri un sottoinsieme di somma data.

Le liste, in tale configurazione, potrebbero essere una, due o molte, ma tutte con lo stesso segno positivo poiché saranno i movimenti che le popolano ad avere segno opportuno.

Il modulo è strettamente collegato alla gestione processi e pratiche. Se un incasso è problematico lo si può assegnare a un ufficio legale con una transizione di stato.

## Script addon

Gli script assistono nell'assegnare uno stesso valore al campo CLUSTERID dei movimenti che si suppone vadano inseriti nella stessa pratica. Negli script il programmatore ha a disposizione un oggetto **\$SEEKER** (da cui “Legend” per il nome del modulo) che fornisce metodi utili allo scopo.

Nel seguente esempio, uno script PHP per un abbinamento elementare basato sull'identità degli importi:

```
<?php
function legendMain(){
    global $SEEKER;

    // L'OGGETTO $SEEKER CONTIENE SOLO LE CHIAVI DEI RECORD
    // FORZO IL CARICAMENTO DI TUTTI I DATI
    $SEEKER->dataload();

    // DETERMINO I DATI DELLE LISTE DENOMINATE Interni ED Esterni NELLA CONFIGURAZIONE
    $int=&$SEEKER->bags["Interni"]["DATA"];
    $est=&$SEEKER->bags["Esterni"]["DATA"];

    // DETERMINO I SEGNI DELLE LISTE
    $intsignum=$SEEKER->bags["Interni"]["SIGNUM"];
    $estsignum=$SEEKER->bags["Esterni"]["SIGNUM"];

    // DETERMINO I GLI IMPORTI DEI MOVIMENTI DELLA LISTA Esterni
    // TENENDO CONTO DEL SEGNO
    $estamount=array();
    foreach($est as $i => $mov){
        $estamount[$i]=$estsignum*$mov["AMOUNT"];
    }

    // INIZIALIZZO LA PROGRESSIONE DI AVANZAMENTO DA COMUNICARE AL CLIENT
    $SEEKER->progressinit(count($estamount));

    // SCANDISCO I MOVIMENTI INTERNI
    foreach($int as $mov){

        // AVANZAMENTO DI UNO NELLA
        $SEEKER->progress();

        // CERCO UN MOVIMENTO ESTERNO CON LO STESSO IMPORTO
        $i=array_search(-$intsignum*$mov["AMOUNT"], $estamount);
        if($i!==false){

            // ASSEGNO UN NUOVO CLUSTERID AL MOVIMENTO INTERNO...
            $CLUSTERID=$SEEKER->clusterize($mov["SYSID"]);

            // ... E LO ASSEGNO ANCHE AL MOVIMENTO ESTERNO DA ABBINARE
            $SEEKER->clusterize($est[$i]["SYSID"], $CLUSTERID);

            // TOLGO IL MOVIMENTO ESTERNO DALLA LISTA
            // PER NON CONSIDERARLO NEI PASSI SUCCESSIVI
            unset($estamount[$i]);
        }
    }
    return true;
}
?>
```

il vettore associativo `$SEEKER->bags` contiene dati e proprietà delle liste di movimenti come sono state definite nella funzione “Configurazioni”.

I metodi di `$SEEKER` sono elencati di seguito; per *lista* si intende un vettore di *SYSID*

(identificatori univoci di sistema, vedere documento *sysid.pdf*); per **tabella** si una matrice di valori; per **mappa** si intende un vettore di trasformazione degli indici che consente di leggere una lista in modo ordinato o per parzializzarla.

Carica in memoria tutti i dati delle liste (se è specificato *\$NAME*, viene caricata soltanto la lista con quel nome):

```
$SEEKER->dataload($NAME="")
```

Assegna ai movimenti, le cui chiavi sono incluse in una lista *\$arrows*, un *CLUSTERID* che può essere passato ovvero creato contestualmente e restituito:

```
$SEEKER->clusterize($arrows, $CLUSTERID="")
```

Utilizza il valore restituito da una funzione (*\$funct*) per aggregare i movimenti di una lista di nome *\$bagname*:

```
$SEEKER->partition($bagname, $funct)
```

Riempie una mappa *\$index* per leggere in modo ordinato una tabella rispetto a una sua colonna:

```
$SEEKER->index($table, $column, &$index, $order=">")
```

Effettua una ricerca su una tabella indicizzata e restituisce la mappa degli elementi trovati:

```
$SEEKER->search($table, $column, $value, $index)
```

Funzioni per effettuare una ricerca numerica mediante il cosiddetto *Algoritmo Zero* ed estrarre un sottoinsieme (mappa) la cui somma eguagli il valore di calibrazione *\$gauge*:

```
$SEEKER->markov_initialize()
```

```
$SEEKER->markov($table, $gauge, $tolerance=0.0001)
```

```
$SEEKER->markov_terminate()
```

Ricerca testi in modo euristico: *\$value* viene cercato nella sulla colonna *\$column* a meno riordinamenti di parole e piccole distanze di *Levenstein*; il risultato viene restituito in una mappa:

```
$SEEKER->smart($table, $column, $value)
```

Restituisce una tabella con i movimenti le cui chiavi sono prese da una lista:

```
$SEEKER->list2table($list)
```

Crea una tabella *\$target* travasando valori dai dati di una tabella *\$source* parzializzati da una mappa *\$map*:

```
$SEEKER->map2table($source, &$target, $map)
```

Riempie un tabella *\$target* con le colonne *SYSID* e *\$column* a partire da *\$source*:

```
$SEEKER->column2table($source, $column, &$target)
```

Restituisce una tabella *\$target* dei soli elementi liberi di un'altra tabella *\$source*:

```
$SEEKER->table2free($source, &$target)
```

Restituisce un vettore associativo con informazioni e totalizzazioni dedotte da una data tabella:

```
$SEEKER->getinfo($table)
```

Normalizza una stringa per creare chiavi da ordinare:

```
$SEEKER->padstring($value, $size=30)
```

Normalizza una data per creare chiavi da ordinare:

```
$SEEKER->paddate($value)
```

Normalizza un numero per creare chiavi da ordinare:

```
$SEEKER->padnumber($value)
```

Normalizza un valore booleano per creare chiavi da ordinare:

```
$SEEKER->padboolean($value)
```

Calcola i giorni fra date “YYYYMMDD”:

```
$SEEKER->datediff($d1, $d2)
```

Aggiunge giorni a una data “YYYYMMDD”:

```
$SEEKER->dateadd($d, $days)
```

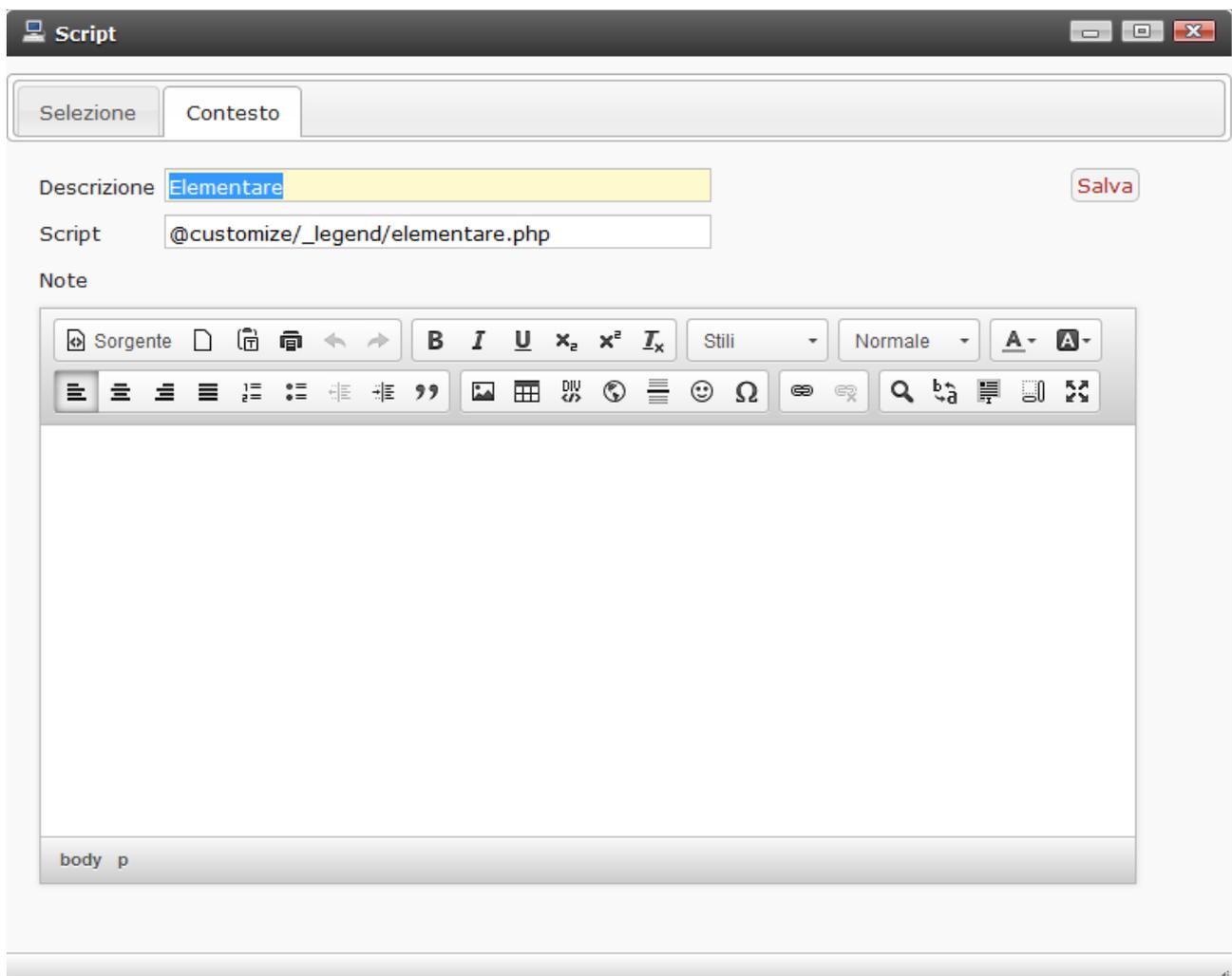
Gestione della progressione di avanzamento:

```
$SEEKER->progressenabled($size=1000)
```

```
$SEEKER->progressinit($total)
```

```
$SEEKER->progress()
```

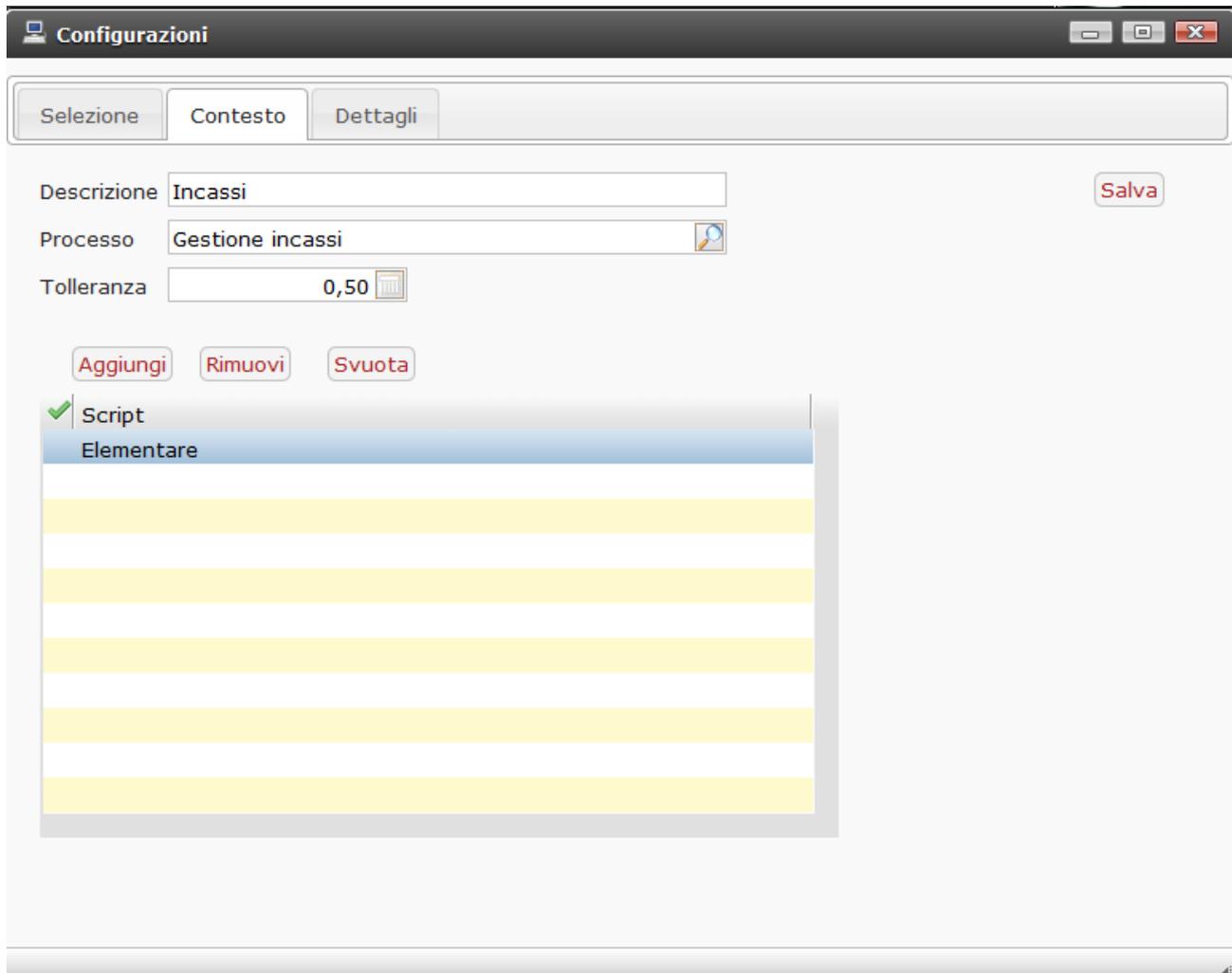
Gli script possono essere censiti sotto il menù “Legend” come nell'esempio:



Il posizionamento predefinito degli script è “@customize/\_legend/”.

# Configurazioni

La funzione consente di definire delle modalità di ricerca. Innanzi tutto possono essere specificati gli script che governano la clusterizzazione:



The screenshot shows a window titled "Configurazioni" with three tabs: "Selezione", "Contesto", and "Dettagli". The "Dettagli" tab is active. The configuration fields are:

- Descrizione: Incassi
- Processo: Gestione incassi
- Tolleranza: 0,50

Below the fields are three buttons: "Aggiungi", "Rimuovi", and "Svuota". A list of scripts is shown below, with a green checkmark next to "Script Elementare". The list contains several empty rows, each with a yellow background.

La tolleranza è un parametro che, all'atto della chiusura di una pratica di abbinamento, impedisce l'operazione se il saldo squadra oltre il valore specificato.

È anche possibile definire numero e caratteristiche delle liste di ricerca (denominate "query"). Tra le proprietà di una lista segnalo **id dati** (*Interni es Esterni* dell'esempio), **vista** (valore facoltativo, utile per avere a disposizione anche campi in JOIN), **select** (elenco parziale di campi da estrarre per ridurre al minimo l'impiego di memoria), **parametri** (vettore JSON che definisce la mascherina di opzioni per l'estrazione dei movimenti), **where** (parametri fissi di estrazione), **colonne** (definizione delle colonne delle liste estratte).

Nell'esempio si noti la condizione `CONSISTENCY=1`; il campo standard `CONSISTENCY` indica il grado di "concretezza" del movimento: se **A** dà 300 euro a **B** il movimento è "effettivo" (`CONSISTENCY=0`); se **A** dà un televisore a **B** del valore di 300 euro, il movimento di 300 euro che rappresenta il totale fattura (movimento interno) è una "equivalenza" (`CONSISTENCY=1`).

Configurazioni

Selezione   Contesto   **Dettagli**

Contesto: Incassi

[Aggiorna](#)

Descrizione
0 Esterni
1 Interni

[Aggiungi query](#)   [Rimuovi query](#)

[Salva query](#)

Descrizione    Ordine    Segno

Id dati    Tipologia    Vista

Select

Parametri (JSON)

```
[
  "OPZIONE1":"C",
  "OPZIONE2":"V"
]
```

Configurazioni

]

Where (SQL)

CONSISTENCY=1

Colonne (JSON)

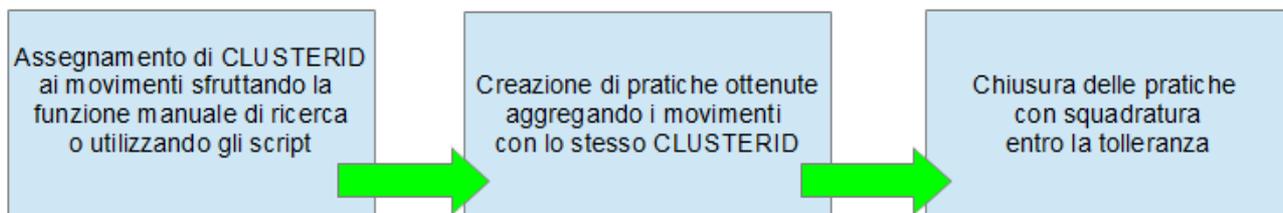
```
[
  {id:"DESCRIPTION", caption:"Descrizione", width:300},
  {id:"AMOUNT", caption:"Importo", width:120, type:"2"},
  {id:"AUXTIME", caption:"Data", width:120, type:"/" }
]
```

[Torna all'elenco](#)

# Clustering

Una volta configurato *Legend* si può passare alla pratica quotidiana con la funzione *Clustering*. Si scelgono la modalità operativa e i parametri di estrazione: ciò determina il caricamento delle liste dei movimenti in sospeso che vanno riconciliati.

L'attività di *matching* si articola in tre fasi:



L'arricchimento di una pratica di abbinamento può essere effettuato su più sessioni di lavoro mano a mano che gli incassi vengono importati nel sistema.

Le pratiche possono essere aperte col modulo **Pi Cubo** (leggere il documento *picubo.pdf*) per perfezionarle con annotazioni e attività.

Pratica	Totale	Inizio
__A0000000077XOP	3' 259,44	21/03/2014
__A0000000077XOS	0,00	21/03/2014

Descrizione	Importo	Data Reg.	Data Val.
__A000000007055Y	1' 878,51	01/06/2010	07/06/2010
00094021	1' 380,93	08/07/2010	08/07/2010

**Movimento:** A000000007055Y      **Descrizione:** \_\_A000000007055Y  
**Conto:** Pippo      **Controparte:**  
**Registrazione:** 01/06/2010      **Valuta:** 07/06/2010  
**Causale:** Pagamento  
**Importo:** 1' 878,51 Euro  
**Cambio:** 0,00      **Segno:** 1

VERSAMENTO TIT CARTA N 94019 DA FILIALE = 00864 VERDELLO





Quest'opera di RODOLFO CALZETTI (<http://www.rudyz.net>) e le idee ivi contenute sono state rilasciate con licenza CREATIVE COMMONS ATTRIBUZIONE - CONDIVIDI ALLO STESSO MODO 3.0 ITALIA.  
Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-sa/3.0/it/> o spedisci una lettera a Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.