

実装！花束問題
＜超高速開発コミュニティ&第44回IT勉強宴会＞
TALONで実装編

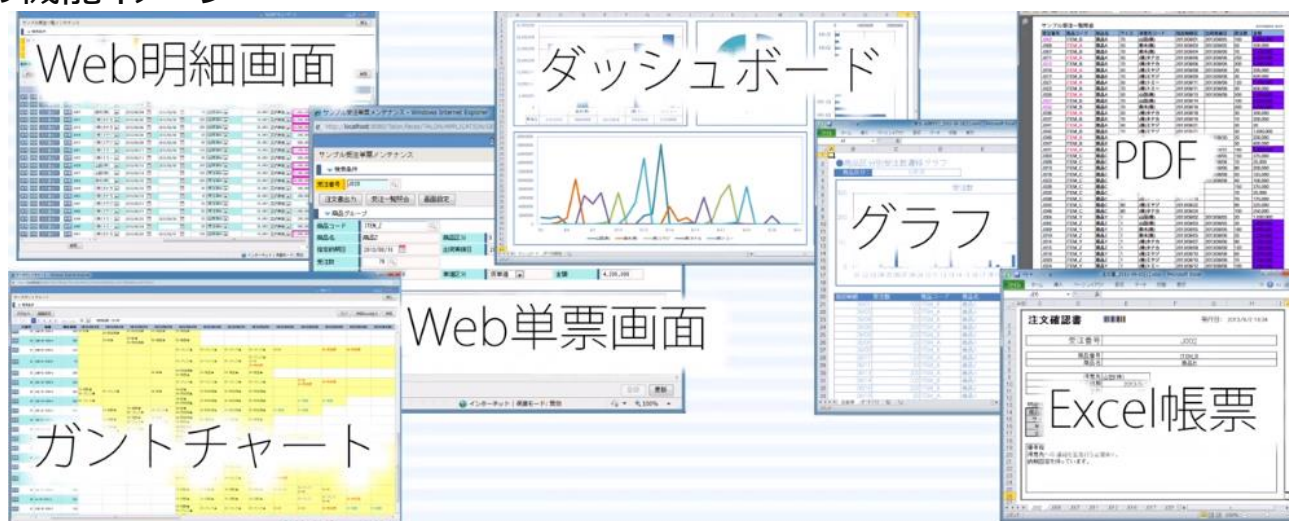
2015年09月19日

- 古関 雄介 38歳
- 超高速開発ツールTALONの開発・販売をしている
(株) HOIPOIという会社を経営しています。
- 私自身は、10年以上、製造業向けシステムを
ひたすら作っていました。
25歳よりPJマネージャとしてDBモデリング、
設計作業をマシンのようにズーッとやっていました。
- 業務システムはDBモデリングが命で、これが最適に
デザインされればPJは成功すると思っ込んでいます。
その結果、データモデル原理主義者へ……。
その思想の元作った製品がTALONです。

02. 超高速開発ツール「TALON」の紹介

- ①WEB画面に取得情報と業務ルールを設定するだけで機能が完成します。
そのため、打ち合わせを行いながら機能構築が可能です。
- ②機能毎にソースコード生成は行わず、WEB画面で設定した情報をデータベースにリポジトリ（設計）情報として格納し、機能が呼び出されるごとに情報を読み込んで画面生成します。
そのため、作った直後から利用可能です。
- ③作られた機能は利用者ごとに改造して自分好みにすることが可能です。
- ④作った機能のリポジトリ情報から常に詳細設計書が出力可能なので、運用している機能と、設計情報の一元管理が可能です。

TALONの機能イメージ



- ・超高速開発コミュニティのモデリング分科会で渡辺さんのX-TEA/受注生産モデルをTALONで実装しました。
(余談ですが、TALON初級者が119機能を約200時間掛けて一応完成しました。)
この企画が縁で今回のイベントにも声を掛けていただきました。

今回花束問題V1.2を実装するという企画を受けて強く感じた事は、

在庫の仕組みを考えるためのギリギリの複雑さである！

ということです。

この問題からは重要な幹部分だけが必要とされ、枝葉の部分はそぎ落とされています。

その為、本質的に必要なテーブル構造だけで、本来非常に複雑な在庫の仕組みが表現できると感じました。

在庫の仕組みは業務システムでは多くの場面で登場します。それは、物品を販売する企業では必ず存在し、利益に直結する重要な仕組みだからです。需要に対して、どのように供給するかで利益が決まります。需要より供給を多く用意すると、

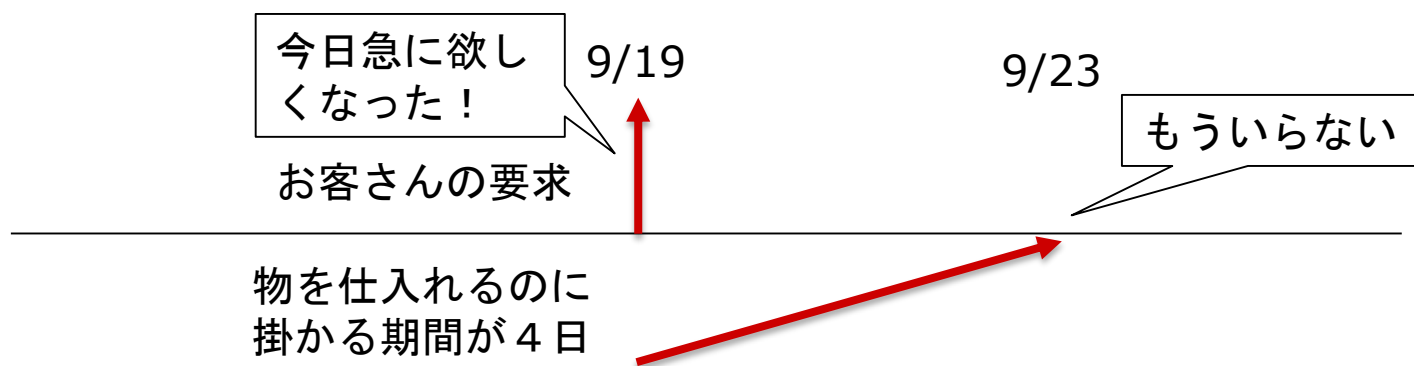
- ・売れなくて廃棄する時にロスになる（原価の高騰）
- ・企業の手持ちキャッシュが減る（C/Fの悪化）

需要より供給が少ないと、

- ・販売チャンスに売ることが出来ない（売上の減少）
- ・ライバル企業に販売されてしまう（顧客の減少）

このバランスを調整するのが在庫です。

在庫管理（需給調整）は難しいとよく言われます。それは、需要タイミングと供給タイミングが違うためです。お客さんはワガママなので、欲しいときにすぐに欲しいです。ですが、欲しいと言われたときに手元に物が無い場合、作ったり仕入れたりする時間が必要です。言い方を替えると需要には時間軸が無く、供給にはあるのです。



この時間ギャップを埋めるのが難しいのではないのでしょうか。

もう一つの理由は、需要と供給の量が違う点です。需要量と同じ量の供給量であれば売れないロスは出ませんが、仕入先の都合や、生産工場の最適生産サイズなどの都合で一致しません。

花束問題はこれらの難しいところは解決しないといけない内容になっています。

しかし、それ以外の部分は加味しなくて良いという割り切りの良い内容なので、本質的な部分だけをモデリング・実装可能です。

これは、在庫、ひいては業務を学ぶまたとない問題だと思います。

与件をダイジェストで確認していきます。

最終目的は、与件にある以下の問題の解決です。

当初は受注も少なく手作業で管理出来ていたが、受注が増えるにつれシステム化の必要性が出てきた。

「新鮮な花を大切な記念日に」を売り文句にしていることもあって、廃棄される在庫が多く、受注の増加にともなって利益が伸びていないため。

上記を考慮して、目的を

「手作業の一部システム化による効率化（生産性向上）
と、花廃棄率の低減（原価削減）による利益向上」

に設定しました。

「手作業の一部システム化による効率化（生産性向上）
と、花廃棄率の低減（コスト削減）による利益向上」

上記目標を行うために、以下を構築します。

- ・顧客注文から発注・仕入、出荷までをシステム化
- ・花廃棄率を低減するために最適な仕入れが出来るための仕組みを構築（見える化）
- ・注文に対して使う花在庫を品質維持期限を加味して最適な物（ぶっちゃけると廃棄ギリギリ）を出荷できるような計算を自動で行う仕組みを構築

商品と在庫の与件を読むと以下のように読み取れます。

- ・商品の花束で、花単品の数と組み合わせを定義。
- ・花束の構成は花単品のみでシングル部品表と同じ。
- ・商品の花束のみが注文され、仕入は花単品のみ。
- ・花単品は品質維持期限があり、仕入先は1社特定。

最もシンプルな形の部品表が定義されており、小売業だけでなく製造業にも対応した作りです。

製品→部品という構成で、製品だけが出荷され、部品だけが仕入れされるというシンプルな形です。

受注・出荷の与件を読むと以下のように読み取れます。

- ・顧客の1注文は1花束のみ。
- ・顧客の注文はお届け日変更が可能だが、出荷できない場合はエラーとする（新規注文時も同様？）
- ・花束を作る加工は一瞬なので加味する必要なし。
- ・出荷は常に1日なので常に1日前に出荷すれば良い。

出荷リードタイムや花束加工に関する部分はほとんど考えなくて良い作りですが、お届け日に花が揃っているかチェックを行う必要がありますし、注文確定後、またお届け日変更されたらその分の花は確保しないとお届け日に届けられないという最悪の事態になります。

発注・入荷の与件を読むと以下のように読み取れます。

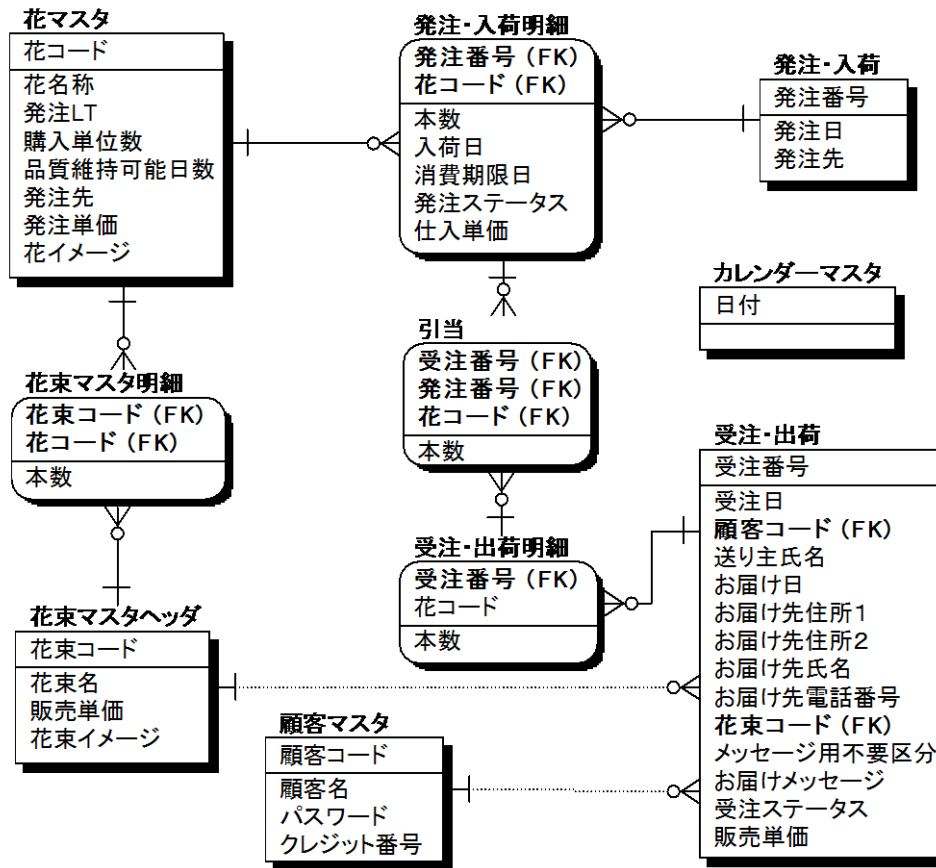
- ・単品ごとにリードタイムが異なる
 - ・リードタイム以降の日であればいくらでも発注可能で、納期通りに必ず入荷される。
 - ・単品ごとに購入可能単位数が決まっている
 - ・花単品は品質維持期限があり、仕入先は1社特定。
- 複数社発注や、分納入荷、納期遅れ対応などを考えなくて良いので発注 = 入荷となり、テーブル構造は非常にシンプルになります。ただし購入の単位数と在庫期限は必要ですし、入荷は単品ごとに入荷日が異なりますので在庫部分は複雑になります。

ここまで確認してきた与件から、以下の方針とします。

- 在庫の本質である需給バランス調整を実現するシステムとするが、この部分以外は与件に忠実なモデル、実装とすることを目指します。
※それでも注文から発注・仕入、出荷まであります。
- データモデルはこれ以上減らせないという所まで絞ったモデルとすることで、業務モデリングに慣れていない、これから学びたいという方が学びやすい物を目指します。

14. 花束問題の与件 データモデル編

ここまでの与件確認を元にしたデータモデルです。



●受注・出荷

分納出荷が存在しないため、受注と出荷のテーブルは同一です。

明細は無くても常にマスタ参照すれば動作しますが、マスタが変更になった場合に困るので注文ごとに保持します。

●発注・入荷

分納や納期遅れが無い前提なので、発注と入荷は同一テーブルです。

さらに、この明細テーブルがロット在庫の役割を果たします。入荷前でも在庫と同じ形で使えるので便利です。

●引当

注文確定時に在庫を自動引当します。別の注文に使われるとまずいので予約として保持します。入荷時にも自動生成

します。この存在が当システムのキモです。

これ以上は削減できないところまで絞っています。

10テーブル50項目になりました。

現在庫などはデータを計算して算出します。廃棄された判断は、引当が作られなかった発注・入荷明細の数量となります。実績はステータス管理のみです。

15. 花束問題のTALONでの機能一覧（子画面は除く）

機能名	機能の概要
花マスター一覧メンテ	花の情報をメンテナンスする。
花マスタ単票メンテ	花の詳細情報をメンテナンスする。
花束マスタメンテ	花束に付与する花単品情報などをメンテナンスする。
顧客マスタメンテ	顧客は自分の情報のみをメンテナンスできる
花束注文	新規注文と注文修正を行う機能。内部で引当を作る
花束注文（管理者用）	花屋さんが使うメンテナンス画面
花束出荷予定一覧	注文情報の確認と、出荷実績へ遷移できる
花束出荷実績入力	出荷を行う。全ての花が引当済でないと出荷できない
加工指示書発行	出荷のための花束加工指示書を発行する
花単品発注入力	発注と発注修正を行う機能。内部で引当を作る
花単品仕入予定一覧	発注情報の確認と仕入実績へ遷移できる
花単品仕入実績入力	入荷を行う。ステータスの変更のみ
花単品在庫一覧	花のサマリされた在庫状況表示。要発注の花が抽出可能。 このままだと廃棄されてしまう情報も抽出可能。 その場合は納期変更などを行えるようにする。
花単品在庫推移画面	花を時系列に在庫推移する状況を表示する。

- 注文（需要）に対して在庫・発注情報（供給）が足りているかは 1 ボタンで確認が出来るようにする。
- 廃棄されそうなデータも簡単に確認でき、発注納期を変更出来るようにする。
- 引当のルールは、
未引当ての在庫・入荷予定に対して、
お届け日に一番近い廃棄日を持ったロットから順に
引当を行っていく。これでも以後の注文により最適では
無くなる可能性があるが、それは人間系で処理できる
ように判断可能な情報提供をする。

17. 画面イメージサンプル（花単品在庫一覧→花単品在庫推移→発注入力）

花単品在庫一覧 - Internet Explorer

http://trialora.talon.jp/hana_jk2H9Xhe/faces/TALON/APPLICATION/GENERALFREE/GENERALFREE.xhtml

花単品在庫一覧

検索条件

花コード

発注先名

未引当残有

A 4 縦 PDF出力

クリア Excel出力 検索

状況一覧

検索結果: 10件

花コード	花名称	現在在庫数	入荷予定数	注文合計 (未出荷)	引当済合計	未引当数 (要発注)	廃棄予定在庫	発注先名
H001	アザミ	10	0	10	0	10	20	仕入先 0 1
H003	スズラン	0	0	20	0	20	0	仕入先 0 2
H006	ヤマゴボウ	0	10	10	5	5	5	仕入先 0 3
H007	ゆり	16	16	3	3	0	29	仕入先 0 4
H008	バラ	14	14	5	5	0	23	仕入先 0 4
H009	彼岸花	0	0	10	0	10	0	仕入先 0 4
H010	チューリップ (アプリコット)	10	10	20	14	6	6	仕入先 0 1
H011	チューリップ (インゼル)	0	20	15	13	2	8	仕入先 0 1
H012	チューリップ (スレクシス)	0	0	15	0	15	0	仕入先 0 2
H013	チューリップ (ピンクダイヤモンド)	0	0	10	0	10	1	仕入先 0 2

廃棄ロスを減らし、出荷遅れが出ないように廃棄状況と未引当ての状況が1アクションで分かり、発注の変更が連続して出来るようになっています。

花単品在庫推移 - Internet Explorer

http://trialora.talon.jp/hana_jk2H9Xhe/faces/TALON/APPLICATION/GENERALFREE/GENERALFREE.xhtml?faces-redirect=false&COLUMN_0=F

花単品在庫推移

検索条件

花コード H010 花名称 チューリップ (アプリコット) 対象日付 (FROM) 2015/09/21 現在在庫 10

A 4 縦 PDF出力

Excel出力 検索

状況一覧

検索結果: 32件

対象日	在庫推移	入荷予定	引当済加工予定	未引当加工予定	廃棄予定
2015/09/21	10	0	0	0	0
2015/09/22	0	0	4	0	6
2015/09/23	4	10	6	2	0
2015/09/24	0	0	4	0	0
2015/09/25	0	0	0	0	0
2015/09/26	0	0	0	0	0
2015/09/27	0	0	0	0	0
2015/09/28	0	0	0	0	0
2015/09/29	0	0	0	4	0
2015/09/30	0	0	0	0	0

花単品発注入力 - Internet Explorer

http://trialora.talon.jp/hana_jk2H9Xhe/faces/TALON/APPLICATION/GENERALFREE/GENERALFREE.xhtml?faces-redirect=false&COLUMN_0=F

花単品発注入力

検索条件

発注番号 PNO150900024

A 4 縦 PDF出力 発注伝票出力

クリア Excel出力 検索

発注・入荷情報

発注日 2015/09/11 発注先名 仕入先 0 1 発注ステータス 発注済

発注・入荷明細情報

検索結果: 1件

花コード	花名称	本数	入荷日	消費期限日	発注ステータス	引当済数	発注LT	購入単位数
H010	チューリップ (アプリコット)	10	2015/09/23	2015/09/28	発注済	10	2	10

発注伝票

発注番号 PNO150900024
 発注先名称 仕入先 0 1
 発注日 2015年9月11日

花コード	花名称	入荷予定日	発注数
H010	チューリップ (アプリコット)	2015年9月23日	10