<센터 현황>

* 1800평이던 군포에서 4800평의 오산 물류센터로 3월에 이전하였음.

동탄에 새로 짓는 물류센터로 이전하기 전에 잠시라도 확장 처리하기 위해 임시로 오산에 오픈

* 기존에 군포의 MPS 3대에서 현재는 8대로 증가하였고 2개의 층으로 이루어짐.

지하 2층은 상온, 지상 7층은 냉장과 냉동으로 운영되는데 주력은 냉동에 특화됨.

7층 -> 냉장과 냉동 상품 보관 rack이 양쪽으로 나뉘어져 있고 MPS는 rack 사이에 위치

* 품질 유지를 위해 MPS는 저온 기준으로 유지
* 상온은 출고 라인이 1개, 저온은 2개(냉장, 냉동으로 운영되지만 유동적으로 사용하기도 함)
* 냉장이 품질에 가장 민감함
* 당일 출고 방식으로 재고량이 많지 않음. 포장 시 냉매제 포함.
* 계절에 따라 센터 자체적으로 테스트를 통해 냉매제의 개수를 정함. (상온에서 최소 24시간 노출된다는 기준)
* 과거 온마트의 가장 큰 문제점은 스티로폼 박스에 대한 수작업이었으나 현재는 자동.
* 오산은 DPS 방식을 사용하는 센터(B2C 고객 대상)

<현재 시스템 운영 현황>

* QMS 시스템: 주문이 생성되면 자동으로 WMS로 올라감.
* 주문이 들어오면 재고가 할당되고 출고 가능 상태로 변경됨.
* 출력 구분: BOX(박스 단위로 출고), SKU(한 가지 단일 상품), MPS(MPS에서 출고), PAT(증정처럼 일정한 패턴을 가지는 주문)

MPS를 제외한 나머지 주문은 MPS를 거치지 않고 별도로 처리 -> 알고리즘에서 제외

* 원하는 패턴을 검색하면 그에 해당하는 주문이 리스트 됨.

그 주문에 대해서만 오더를 내릴 수 있음.

* 1차적으로 재고가 할당이 되고 주문이 출고 가능 상태로 바뀌면 패턴으로 조회 가능

패턴이 일정 % 이상이면 자동으로 잡아 줌.

* MPS에 전송할 때 차수를 나눠서 전송

각 MPS 호기 별로 주문 전송

주문을 넣은 순서를 전표번호라 하고 이 순서대로 전송됨.

* 창고구분: 각 호기 별로 유형이 나뉘어져 있어서 차수 생성에 따라 각 호기로 주문이 전송됨.
* 패턴을 가지는 특정 상품들은 MPS를 거치지 않기 때문에 검수, 포장에서 병목현상 발생

따라서 다른 주문을 할당 할 수 없음

* E-COMMERCE에서 주문의 형태가 계속해서 변동됨. 계절성 등으로 상품의 변화가 자주 일어 남. 그래서 특정 상품을 MPS에 고정할 수가 없음.
* 출고구분에서 MPS 타입에 있어서 주문량이 다른데 지금은 접수순으로 할당되어 있음.
* 주문이 업체 별로 들어오기 때문에 센터 별로 나눠서 처리

EX) 11번가, 쿠팡, 옥션 등

배송은 같이 할 수도 있음

* WMS에서 MPS로 주문을 전송하면 주문이 내려옴 한꺼번에 내려오지는 않음

<Total Picking>

* 처음 작업할 때는 보관된 곳에서 물건을 가져와서 Cell에 채워서 작업 시작(Total Picking)
* 많은 조합의 주문을 먼저 처리하기 위해 그 조합의 물건들을 먼저 셀에 채움.

다른 조합이 들어가면 나중에 처리.

* 가장 많은 조합의 상품들을 전체 상품에서 찾아내는게 어려움

작업자들이 본인이 편한대로 셀에 채우기 때문에 가장 많은 조합의 상품들을 바로 앞에 채워 놓기가 힘듦

* MPS는 상품 적재가 만족되면 그 순간 주문의 순서를 결정

상품이 적재되면 그 상품에 해당되는 주문이 또 할당됨.

* Q-알고리즘: 어떤 상품을 먼저 Cell에 적재하냐에 따라 작업 주문의 순서 결정(상품이 먼저 다 채워지는 주문이 먼저 순서가 잡힘)
* TV 상황판에 작업자들에게 어떤 상품을 가져오라고 알려줌

하지만 작업하면서 쳐다보기가 어려움

* 오늘의 프로모션 제품들을 찾아서 먼저 배치.

프로모션이 걸리면 그 제품들의 주문량이 증가하고 사람이 이를 인식할 수 있지만 시스템은 알기 어려움.

* 가장 많은 주문의 상품들을 분산 적재해서 한 사람에게 몰리지 않도록 함

<검수 포장 방식>

* 주문에 따라 박스의 크기를 결정해서 박스를 MPS로 보냄
* 단수(햇반 1개 주문), 단포(햇반 10개 주문), 복수(여러가지를 여러 개 주문)
* 단수는 한 피스만 주문한 사람들만 모아서 상품들을 그 양만큼 가져와서 물건을 하나 집어서 상자에 담는다.
* 단포는 a 상품을 바코드 찍으면 그 상품을 주문한 주문들을 보고 개수 확인, 같은 개수 별로 분류가 됨. -> 2개짜리만 뽑고 3개짜리만 뽑고, 각자 개수별로 포장
* 복수는 똑같은 주문을 한 사람들, 똑같은 종류의 상품 주문 먼저 처리

주문의 조합을 보고 같은 조합의 주문들을 같이 처리

많이 나가는 상품들이 있고 그것들만 사는 사람들이 많음, 그 상품에 대한 주문들이 먼저 나가도록 처리

* 20% SKU가 전체 주문의 50%이상을 차지함
* 중간에 상품 Picking이 완료되면 중간에서 빼냄

<알고리즘에 대한 전문가 의견>

* 상온, 냉장, 냉동 중 하나의 온도만 정해서 알고리즘을 개발해도 괜찮을 듯
* 실무자들의 의견을 들어보고 프로젝트의 SCOPE 정하기
* 프로젝트의 성과를 생산성이나 작업자, 직원의 업무 강도 변화 등으로 평가하는 기준을 잡아야 함.

작업 시작 시간부터 종료시간까지의 투입된 작업자 수 대비 생산성 향상으로 평가해도 좋음.

현재 생산성 평가 방식은 센터장님께 여쭤봐야 함.

* 프로그래밍 코드보다도 SUDO 코드처럼 우리가 이해할 수 있는 언어로 알고리즘을 짜는게 중요.
* 많은 조합의 상품들은 MPS 상에 적재 안 할 수 있도록 유사한 주문 패턴을 찾는 것이 중요

유사주문의 패턴을 볼 때 몇 프로까지 확인하는지 알고리즘이 필요.

* 시뮬레이션 할 필요가 있음.
* 검수 포장 단계에서도 개선되었으면 함.
* 신병철님 曰: 차수에 따라 주문을 보고 훨씬 효율적일 거 같은 차수를 먼저 해결하면 좋을 거 같음 -> 하지만 이렇게 하려면 1차수, 2차수 주문에 대한 수량을 전체 확인해야 함.
* 온라인 센터의 전체 주문량 중에서 그 상품으로 구성된 주문이 일정 % 이상인지 그 기준 값은 센터마다 다름.

사람이 이 기준 값을 시스템에 입력만 해줘도 일이 쉬워질 수 있음.

그 일을 매일 아침마다 해야 하지만 사실상 어려움.

따라서 시스템에서 자동으로 받아낼 수 있으면 좋을 듯.

* A, B, C 상품이 수량이 많을 때 셀에 모아주는게 효율적일지 띄엄띄엄 놓아주는게 효율적일지는 생각해보아야 함