



# مدیریت عملیات

دکتر مهدی شیخزاده

زمستان ۹۴ - بهار ۹۵

نگارنده: محمد قربانیان

با تشکر از سجاد احمدی

به نام خداوند جان و خرد

## فهرست مطالب

۱	معرفی و بررسی اهمیت مدیریت عملیات
۳	مقدمات مدیریت عملیات
۴	نوع شناسی (typology) برای عملیات
۴	ارزیابی عملیات
۵	روندها
۵	روند: هر چه زمان می گذرد، فضا به سمت خدمات می رود.
۵	روند: جهانی شدن (Globalization)
۶	روند: کیفیت بهتر محیط زیست:
۶	روند: مدیریت cross-functional :
۶	روند: تجارت الکترونیک
۶	روند: مدیریت زنجیره تأمین (SCM):
۶	روندهای حاکم بر مدیریت زنجیره تأمین:
۷	Strategy Fit
۷	قدم اول: درک عدم قطعیت مشتریان و زنجیره ی تأمین
۹	قدم دوم: فهمیدن زنجیره ی تأمین
۱۰	قدم سوم: Achieving Strategic Fit
۱۲	مدیریت خدمات
۱۳	Service Classification
۱۳	چالش های مدیران خدمات
۱۵	Service package
۱۶	ارزیابی خدمات
۱۸	ویژگی های خدمت

۱۹	..... خدمت یا محصول؟
۱۹	..... Capacity Management
۱۹	..... Demand Management
۲۱	..... performance measurement
۲۲	..... مقایسه‌ی خدمات و تولید از لحاظ فرآیند
۲۳	..... layout
۲۴	..... (Purchasing) Material Management
	تفاوت بین purchasing, Procurement, Buying, Sourcing و Supplying
۲۷	
۲۷	..... پارامترهای اثر گذار
۲۹	..... پارامترهای اثر گذار (از دید خارجی)
۲۹	..... پارامتر های موثر (داخلی)
۲۹	..... ساخت یا خرید:
۳۱	..... استراتژی برای buy یا hybrid
۳۲	..... Design Collaboration
۳۲	..... Strategic Issues- Local vs. Global View
۳۵	..... Purchasing Paradaigms
۴۳	..... IT Value Chain
۴۴	..... Inventory Management
۴۵	..... لایه‌های مدیریت موجودی
۴۵	..... انواع موجودی:
۴۶	..... The role of cyle inventory in Supply Chain
۴۹	..... ABC classification
۵۰	..... Impact of Aggregation on safety inventory
۵۲	..... multii-echelon

۵۳	..... Typical Inventory Process
۵۴	.....Closed Loop Process in Practice
۵۵	..... مدیریت تولید
۵۷	.....Product Variety
۵۸	..... Market and Socieity needs
۵۸	..... Dedicated Manufacturing lines
۵۸	..... Planning
۶۰	..... Some planning guidelines
۶۱	.....ERP (Enterprise Resource Planning) <--> APS (Advanced Planning System)
۶۱	..... پیچیدگی های حوزه ی برنامه ریزی تولید
۶۳	..... Demand priorities
۶۳	.....Dependent vs. Independent demand
۶۴	..... Date effective
۶۴	..... Frozen window
۶۴	..... Holiday Calendar
۶۴	.....Max lateness
۶۵	..... Max delivery shipment
۶۵	.....Build Ahead limit
۶۵	..... Fair-share
۶۵	..... Constrained vs unconstrained planning
۶۶	..... Production Planning System
۶۶	..... چالش ها در راستای برنامه ریزی تولید
۶۶	..... : permanant Demand-Supply Balance چالش اول
۶۷	.....Ensure an Optimal D/S Balance : چالش دوم
۷۲	..... Traditional Production Planning - MRP

۷۳	.....	لایه‌های مختلف برنامه‌ریزی برای تولید
۷۴	.....	Master Planning
۷۴	.....	Distribution
۷۶	.....	distribution roles
۷۷	.....	نقش اول: Combining right portfolio of products
۷۷	.....	نقش دوم: Delivery of products in a right time to a right place
۷۷	.....	نقش سوم: Market assessment
۷۷	.....	سطح‌های توزیع:
۷۷	.....	سطح اول: Network Desing
۷۹	.....	سطح دوم: Planning & Schedueling
۷۹	.....	سطح سوم: Excecution
۸۰	.....	Advanced Planning Systems
۸۶	.....	Demand Planning
۸۸	.....	Inventory Planning
۸۹	.....	Master Planning
۹۱	.....	Demand Fulfillment

## معرفی و بررسی اهمیت مدیریت عملیات

اهمیت درس مدیریت عملیات بسیار زیاد است. هر کس عملیات بداند، تقریباً یک دید کلی از هر حوزه به دست می‌آورد. می‌خواهد آن درس، HR باشد، بازاریابی باشد و... . عملیات از جنبه‌ی کلان تا ریز را شامل می‌شود. همیشه دو چشم‌انداز لوکال و گلوبال وجود دارد. تصمیمات ممکن است که از این دو دیدگاه دچار تناقض باشد. مثلاً از جنبه‌ی مالی، ممکن است که یک فرد پول نقد داشته باشد، خودرو داشته باشد، خانه داشته باشد. باید دید از نگاه لوکال چه کار می‌کند؟ مثلاً پول نقدش را چه کند؟ توی بانک بگذارد؟ با خودروش چه کند؟ آیا آن را تعمیر کند یا عوض کند؟

در صورتی که به صورت گلوبال نگاه شود، مجموعاً شرایط بررسی می‌شود. نگاه لوکال و گلوبال معمولاً دو دستاورد مختلف دارند. این خیلی مهم است که ببینیم که گلوبالی تصمیم می‌گیریم یا لوکالی؟

تصمیمات چه قدر centralized می‌باشد و چه قدر decentralized است؟ مثلاً وقتی بودجه را به استان‌ها بدهیم و بگویم هر کاری می‌خواهید بکنید، بکنید. یا به دیپارتمان‌ها بدهیم و هر کدام به صورت مستقل هر کاری که می‌خواهند انجام بدهند، انجام دهند. این نوع بر روی لوکال بودن تصمیمات تأثیر مثبت می‌گذارد. دیدگاه centralized که باشد، تصمیمات هم گلوبال می‌باشند. درس مدیریت عملیات، به طور سنتی راجع به موجودی و لجستیک صحبت می‌کرد و با توسعه‌ی فناوری اطلاعات توان تصمیم‌گیری راجع به مسائل کلی‌تر شد و تقریباً علم زنجیره‌ی تأمین شکل گرفت.

در دیدگاه گلوبال و لوکال در پزشکی می‌توان به این مثال اشاره کرد که وقتی دردی داشته باشید، اگر پیش دندان‌پزشک بروید، یک نظر می‌دهد. دکتر معده یک نظر می‌دهد ولی اگر بخواهند به طور گلوبال نگاه کنند، می‌گویند که این‌ها همه به هم مربوط است و ممکن است که به اعصاب ربط داشته باشد.

این رویکرد را به همه جا می‌توان تعمیم داد. مثلاً در حادثه‌ی یازده سپتامبر، CIA، FBI را متهم می‌کرد که تو کارت را درست انجام نداده‌ای. تو چرا اجازه داده‌ای که یک فرد بدون مدرک، خلبانی کند؟ FBI هم جواب می‌داد که تو خارجی‌ها را درست بررسی نکرده‌ای. بعد از آن HOME LAND SECURITY شکل گرفت که وظیفه داشت، روندها را به طور گلوبال نگاه کند.

در حوزه‌ی مشاوره، هم چنین رویکردی مطرح است. شرکت‌های مشاوره، ادعا می‌کنند که ما در یک شاخه تخصص داریم. مثلاً در استراتژی تخصص داریم. این ادعاها برای گذشته بود. ولی امروزه روند به سمت گلوبال حرکت کرده است. مثلاً من اگر بخواهم زنجیره تأمین ایران خودرو را بررسی کنم، نمی‌توانم از مسائل مالی و منابع انسانی و ... چشم‌پوشی کنم و بگویم این مشکل شما است. شرکت مشاوره می‌گوید، من اگر در استراتژی مشاوره می‌دهم، جنبه‌های دیگر را هم دیده‌ام.

مدیریت عملیات به طور سنتی راجع به موجودی و لجستیک صحبت می‌کرد و با توسعه‌ی فناوری اطلاعات و افزایش توان تصمیم‌گیری علم زنجیره‌ی تأمین شکل گرفت.

Global view as much as possible and Decentralized as much as needed

نگاه integrated روزانه، در حال گسترش است. حتی در مسائل معرفتی، چنین چیزی دیده می شود. اگر این نگاه integrated نباشد، برای جهان نسخه های غیرمفید پیچیده می شود.

به نظر می رسد، تصمیم centralized به گلوبال نزدیک است. شاید مفید باشد که این طور تصمیم گیری کنیم ولی اگر میزان آن را زیاد کنیم، به مشکل بر می خوریم. اگر همه چیز بخواهد centralized باشد، حساسیت ها بیش از حد می شود و آدم ها در سازمان می گویند که پس ما در سازمان، عروسی بیش نیستیم و شما تصمیم می گیرین که ما چه کاری انجام دهیم. این centralized بودن، نباید تا حدی باشد که نیروی انسانی، انگیزه اش را از دست بدهد.

نگاه بالا به پایین و پایین به بالا: شرکت سامسونگ، اگر بخواهیم به طور دیویژنال نگاه کنیم و هر کدام از دیویژن ها را بررسی کنیم. یک ساختار بالا به پایین پیدا می کنیم. مثلاً ممکن است که ما در بازار سهام، بخواهیم که جایگاه خود را حفظ کنیم، باید افزایش درآمدی پانزده درصدی را در نظر بگیریم. از دید بالا به پایین مشخص می شود که مثلاً برای گلگسی چه مقدار باید افزایش فروش داشته باشیم. نگاه پایین به بالا، از پایین شروع می شود، مثلاً می رویم به دپارتمان فروش می گوییم، شما چه قدر می توانید افزایش فروش دهید؟ در نهایت از پایین به بالا، برای بالا یک نتیجه گیری خاصی انجام می شود. به طور عموم، دیدگاه پایین به بالا و بالا به پایین، انسجام کلی ندارند. این نگاه، در دیدگاه معرفتی هم وجود دارد. مثلاً ما انسان ها مختار هستیم. عالم هم پر از randomness می باشد. از منظر بالا به پایین، (خدایی) همه چیز علت و معلول است. چه این رعد و برق و چه این زلزله، همگی شامل یک نظام علت و معلولی است. اشکال این جاست که این دو نگاه را با هم قاطی کنیم. از منظر بالا به پایین، آدم موجود جبری است و از پایین به بالا، انسان کاملاً مختار است و اختلاف فلاسفه بر سر همین موضوع است. این نگاه تنها به درس مدیریت عملیات ختم نمی شود.

نگاه کوتاه مدت و بلند مدت: در کشور ما، دیدگاه اکثراً کوتاه مدت بوده است. اکثر وزارت خانه ها نیروی انسانی ندارند، چون هر کس که می آمده است، فقط به فکر دو سه سال خود بوده است. در نگاه کوتاه مدت، هزینه های transaction زیاد است چون همیشه transaction در حال رخ دادن است. نگاه کوتاه مدت از جنس پول نقد است و نگاه بلند مدت از دید سرمایه گذاری است. در درس مدیریت عملیات، به طور گسترده ای با این مباحث درگیر هستیم.

کتاب خوب راجع به این درس به طور کلی پیدا نمی شود. استاد معتقد است که بسیاری از مباحثی که مطرح می شود، ناشی از تجربه است و از کتاب خاصی ناشی نمی شود. استاد آن چه در عمل تجربه داشته است را منتقل می کند.

راجع به استراتژی به طور مختصر در درس بیان می شود. مدیریت خدمات به طور گسترده تری بیان می شود. چون دیگر به ندرت دیده می شود که یک محصول بدون خدمات ارائه شود. خدمت، نیازمند تکنولوژی است. به

نگاه کوتاه مدت از جنس پول نقد است و نگاه بلند مدت از جنس سرمایه گذاری است.

طور مفصل راجع به procurement بحث می‌شود. جایی است که به شدت پول به خاطر آن از سازمان خرج می‌شود. راجع به موجودی بحث می‌شود. مدیریت تولید و ... هم بررسی می‌شود.

## مقدمات مدیریت عملیات

در دنیای عمل، هر بنگاه و شرکتی یک خروجی دارد که یا محصول یا خدمت است. به طور مثال، یک شرکت مشاوره و یک دانشکده، خدمت را عرضه می‌کند. به طور کلی هر کاری که در سازمان انجام می‌شود و به نوعی به خروجی مربوط است، عملیات است. فعالیت‌های اصلی سازمان، عملیات است. اگر یک سازمان، خدمت آموزش ارائه می‌دهد، منابع و HR و دیگر منابعی که دارد، نقش حمایت‌کننده برای آموزش دارد. عملیات، فلسفه‌ی وجودی یک بنگاه است. اگر مواد اولیه برای تولید خرید شود یا نیروی انسانی برای آموزش استخدام شود مربوط به عملیات است. هر آنچه که مربوط به خروجی سازمان است، عملیات است و موارد دیگر مانند منابع انسانی، جزء بخش حمایتی می‌شود. امروزه خدمات این قدر گسترده شده‌اند که کالا با خدمات، باندل می‌شوند.

وقتی صحبت از کالا یا خدمت می‌شود، یک طیفی وجود دارد. اگر یک محصول خریده شود و دیگر فرآیندی مانند تعمیرات بعد از آن نباشد، به طور محض کالا است.

یک بنگاه، یک سری ورودی دارد و یک سری خروجی. آنچه بین ورودی و خروجی رخ می‌دهد، فرآیندهای عملیات است. مواردی نظیر طراحی، برنامه‌ریزی جزء این موارد حساب می‌شود. برای استراتژی‌های عملیات، یک سری حلقه‌هایی وجود دارد که با دیگر حوزه‌های سازمان مرتبط است. ترجمان استراتژی‌های کلی سازمان، در هر بخش، استراتژی آن بخش را مشخص می‌کند. قلب تپنده‌ی سازمان مربوط به فعالیت‌های عملیات است. مثلاً نیروی انسانی، مشخص می‌کند که چه میزان افراد می‌توانند فعالیت کنند. به طور مثال اگر لازم باشد که یک برنامه‌ای برای افزایش انگیزش کارکنان چیده شود، ممکن است که یک راه حل این باشد که با تغییر در عملیات، بار سنگین از روی افراد برداشته شود تا اصطکاک دائم بین افراد کم شود.

بازاریابی جنبه‌ی تقاضا را بررسی می‌کند. عملیات هم جنبه‌ی تقاضا را پاسخ می‌دهد و این دو حوزه با هم تنیده شده‌اند. نمی‌توان با تبلیغات تقاضا را زیاد کرد در صورتی که عملیات نتواند پاسخ این تقاضا را بدهد. در حوزه‌ی طراحی و توسعه‌ی محصول، تیم طراحی باید با تیم عملیات در تعامل باشند.

درک زنجیره‌ی تأمین بسیار مفید است. به طور مثال در فضای خرید سهام، وقتی قیمت نفت پایین و بالا شود، با توجه به زنجیره‌ی تأمین، خیلی از موارد را می‌شود درک کرد و راجع به سهام‌های شرکت‌ها، تصمیم‌گیری کرد.

عملیات، فلسفه‌ی وجودی یک بنگاه است.

ترجمان استراتژی‌های کلی سازمان، در هر بخش، استراتژی آن بخش را مشخص می‌کند.

اگر عملیات را افراد خوب بلد باشند، می‌توانند بازیگر بهتری در دیگر حوزه‌ها باشند.

عملیات به فرآیند فعالیت‌هایی که مربوط به خروجی سازمان است، مربوط می‌شود.



### نوع شناسی (typology) برای عملیات

هر چه به سمت بالادست صنایع برویم، به نوعی به منابع طبیعی نزدیک می‌شویم. هر چه به سمت صنایع پایین دست بیایم، جنس و تایپولوژی تغییر می‌کند. به طور مثال در پتروشیمی‌ها، سرمایه‌گذاری‌ها سنگین است و تنها چند محصول با حجم زیاد وجود دارد. ولی وقتی پایین‌تر بیایم، تعداد محصولات زیادی با حجم کمتری وجود دارد و سرمایه‌گذاری هم کم‌تر است. utilization (بهره‌مندی) منابع در صنایع بالادست مهم است. در صنایع پایین دست، شرکت‌های زیرپله‌ای material insentive هستند که چند چیز را با هم تولید می‌کنند. در صنعت نیمه‌رساناها، تایوان رتبه‌ی اول را دارد. در صنایع بالادستی تولید ویفرها و چیپ‌ها رخ می‌دهد که سرمایه‌گذاری بالا دارد. در صنایع پایین دستی این موارد، می‌توان به سامسونگ و دل اشاره کرد، که تنها به مونتاژ می‌پردازند.

به طور معمول، صنایع که حجم بالایی دارند، تنوع کمتری دارند و استاندارد و روتین بیشتری دارند. علاوه بر این تغییرات در تقاضا کم است. به طور مثال مصرف کل کشور در انرژی و فلزات، زیاد تغییر نمی‌کند. تغییرات شدید در تقاضا مربوط به محصولات پایین دست است. این که محصول در آخر به دست چه کسی می‌رسد، در visibility مشخص می‌شود. به طور مثال در بحث خودرو یک فرد خودرو را به طور سفارشی، سفارش می‌دهد و visibility زیاد است.

به طور مثال، در صنعت دارو، بعضی از بنگاه‌ها، برای دیگر بنگاه‌ها تولید می‌کنند و خودشان بالادستی هستند. شرکت‌هایی وجود دارند که خود شرکت‌های بیمه و بانک‌ها را بیمه می‌کنند. در واقع این شرکت‌ها، جزء صنعت بالا دستی حساب می‌شوند.

### ارزیابی عملیات

هر کاری که انجام می‌شود، بهتر است که اندازه‌گیری شود. اگر کاری اندازه‌گیری نشود، می‌توان گفت که مدیریت رخ نمی‌دهد. به طور مثال، آیا گمرک از شرکت هیوندای می‌پرسد، شما چند ماشین وارد کرده‌اید؟ (اندازه‌گیری وجود ندارد)

**هزینه**، یکی از مهم‌ترین شاخص‌های اندازه‌گیری است. کاری که انجام می‌شود، چه قدر هزینه می‌برد؟ در کشور ما، توجه به این موضوع نیاز است. تولید خودرو و گندم در داخل کشور، به چه هزینه‌ای رخ می‌دهد؟ در عملیات، با کاری انجام شود که در آن core competency وجود دارد و دیگر کارها باید، برونسپاری انجام شود. به طور مثال، بوئینگ و ایرباس، بردهای الکترونیکی‌شان را به ژاپن برای تولید می‌سپارند. آیا این وابستگی است؟ هنر در جهان جدید این است که بتوان تعامل داشت.

چه **سرعتی**؟ این کالا که با ده دلار تولید می‌شود، دو روزه تولید می‌شود یا دو هفته‌ای؟

در یک سازمان کسی باید استراتژیست شود که موی سپید کرده‌است و دقیق بداند که پیامد نسخه‌اش برای سازمان چه خواهد بود.

در هر شرکتی که وارد می‌شوید، ابتدا بفهمید در کجای صنعت و طیف‌ها قرار دارد؟ بالا دست؟ پایین دست؟

کالا با چه کیفیتی تولید می‌شود؟

چه قدر انعطاف وجود دارد؟ اگر در حال تولید لامپ صد وات می باشد، اگر تقاضا عوض شد، آیا بنگاه می‌تواند سریع لامپ شصت وات تولید کند؟

چه قدر dependability وجود دارد؟ وقتی بنگاه تعهد می‌کند که دو روزه یک کالا را تحویل دهد، آیا در عمل هم دو روزه تحویل می‌دهد یا دو ماه؟

بحث total cost of ownership، مطرح می‌شود. به طور مثال ممکن است که داشتن یک ماشین بیست میلیون به خاطر هزینه‌های بعدی‌اش مانند قطعات، کم‌تر به صرف باشد تا داشتن یک ماشین پنجاه میلیونی.

این پارامترها از هم مستقل هستند به خاطر اینکه دیمانسیون آن‌ها متفاوت است ولی با همدیگر کورلیشن دارند.

این درس به صورت ماژولار ارائه می‌شود. در بعضی از مدرسه‌های کسب و کار، رویکرد ماژولار نمی‌باشد و ابتدا مثلا استراتژی بررسی می‌شود و بعد از آن عملکردها بررسی می‌شود.

## روندها

برای یک روند، غالب بودن آن مهم است. یک روند دیر یا زود، در هر کشوری وارد می‌شود.

**روند: هر چه زمان می‌گذرد، فضا به سمت خدمات می‌رود.** ابتدا زمان کشاورزی بود، سپس تولید بود و امروزه در کنار تولید، خدمات دارد جای خود را باز می‌کند. مثلا حتی برای خرید یک میز، خدماتی مثل تحویل و تعمیرات وجود دارد و تولید با خدمات باندل می‌شود. حوزه‌ی خدمات برای کشورهایی مثل ایران که صاحب تکنولوژی نیستند، بسیار مناسب است. فرض کنید که با بازشدن درهای کشور، تکنولوژی‌ها وارد کشور شوند، این تکنولوژی‌ها نیازمند خدمات هستند. همیشه محصول مهم نیست. مدیرانی که عقلشان در چشمشان است، موارد قابل مشاهده را فقط در نظر می‌گیرند. به طور مثال اگر به آن‌ها گفته شود که فلان میز را باید بخرید برای شرکت، قبول می‌کنند ولی برای چیزهای نرم، مثل رضایت‌مندی مشتریان در سازمان چون محسوس نیست، مقاومت می‌کنند.

**روند: جهانی‌شدن (Globalization):** شرکت‌ها به این باور رسیده‌اند که هر آنچه را که قابلیت ندارند، خودشان انجام ندهند و بدهند به دیگران. این، کار جهانی‌شدن را شکل داده است. به طور مثال، قبلا در یک شرکت، همه چیز را در داخل شرکت انجام می‌دادند. تمام کدهای بخش IT در شرکت انجام می‌شد و حتی آموزش نیروی انسانی هم در داخل شرکت انجام می‌شد. اکنون، گفته می‌شود اگر یک کار را خودتان بلدید، انجام دهید ولی اگر بلد نیستید، بهتر است که بدهید دیگران انجام دهند. ممکن است که گفته شود، این کار

حوزه‌ی خدمات برای کشورهایی مثل ایران که صاحب تکنولوژی نیستند، بسیار مناسب است.

وابستگی می‌آورد. الزاماً اینگونه نیست و فقط برای بعضی از آیتم‌های استراتژیک این طور است. با توجه به جهانی‌شدن، مناطق جغرافیایی گسترش یافته‌اند. شفافیت اطلاعات زیاد شده‌اند. به طور مثال برای خرید یک محصول پزشکی قبلاً می‌بایست به خیابان جمهوری فقط رجوع می‌کردیم، امروزه با جستجو در اینترنت ممکن است که یک محصول را در تایوان، بپسندیم که برای ما مناسب‌تر است و آن را با حتی قیمت کمتری بخریم.

**روند: کیفیت بهتر محیط زیست:** قبلاً، هر جور که بود تولید می‌کردیم و هوا را آلوده می‌کردیم و محدودیتی نبود. امروزه، بشر به این باور رسیده است که باید این موضوع را کنترل کند. این‌ها در دنیای زنجیره تأمین پیامد دارد. این موضوع ابعاد مختلفی دارد. هم می‌تواند، بار مثبت داشته باشد هم بار منفی. به طور مثال، ممکن است برای نجات جان یک حیوان، چندبرابر انرژی که برای نجات او ارزش دارد، انرژی مصرف شود. مثلاً، چینی‌ها برای رقابت در بازار جهانی، بعضاً مثل برده دارند کار می‌کنند که این‌ها در جهان جدید، مشکل دارد.

**روند: مدیریت cross-functional :** نگاه integrated به مرور زمان، دارد پررنگ می‌شود. نگاه Business Process Reengineering مطرح است.

**روند: تجارت الکترونیک:** قبلاً، تعداد زیادی واسطه بین محصولات بود.

**روند: مدیریت زنجیره تأمین (SCM):** تفاوت مدیریت زنجیره تأمین و عملیات اینجا مشخص می‌شود: اگر داخل شرکت را بخواهیم مدیریت کنیم که چه چیزی کجا باشد، مدیریت عملیات است ولی وقتی فراتر از شرکت می‌رویم این که، در شرکت‌های بالادست و پایین دست چگونه تعامل کنیم، بحث زنجیره تأمین پیش می‌آید.

#### روندهای حاکم بر مدیریت زنجیره تأمین:

اگر فضای کسب و کار را از یک منظر دیگر نگاه کنیم، پیچیدگی در حال بالا رفتن است و تکنولوژی این اجازه را می‌دهد، که پیچیده‌تر تولید شود. انتظارات مشتریان بالا رفته است و شرکت‌ها عملاً در تولیدات به صورت سفارشی پیش می‌روند. تنوع محصولات بالا رفته است. پیامدهای این روندها این را نشان می‌دهد که نیاز به coordination زیاد شده است. شرکت‌ها قبلاً، تمام فعالیت خود را در داخل انجام می‌دادند و کل کار را از الف تا ی انجام می‌دادند. در جهان جدید، کسی برنده است که بتواند تعاملات را به خوبی مدیریت کند. به طور مثال در دانشکده خوب است که از بیرون هم دعوت کنیم و تعامل داشته باشیم. چند شرکت داخلی در ایران وجود دارد که قدمت صد ساله دارد؟ بین مالکیت و مدیریت تفاوت قائل شود؟ هیئت مدیره‌ی شرکت‌های داخلی، اکثر خانوادگی مثل دختر و عروس است. تعامل کردن از دیدگاه تئوری بازی‌ها نیز قابل بحث است. در مثال مرتبط با تئوری بازی‌ها، می‌توان به prisoners dilemma اشاره کرد که اگر به طور تکرار انجام شود و بازی به طور repeated game باشد، دیده می‌شود که بهتر است تعامل انجام شود. ایران خودرو مثلاً دو مدل ذهنی را در رابطه با قطعه‌سازان می‌تواند در پیش بگیرد: ۱- آن‌ها را له کنم و به جای پرداخت دو ماه، چهار ماهه پرداخت کنم. ۲- به آن‌ها کمک کنم که رشد کنند. مثال دیگر راجع به خودروسازها: یکی از خودروسازهای داخلی (کرمان خودرو) از نظر تعداد افراد بسیار محدود است ولی کارهای خوبی انجام می‌دهد. در حوزه‌ی نیمه رساناها:

بعضی از شرکت‌ها هستند که کلاً ده نفر مهندس طراح هستند ولی جریان کار را مدیریت می‌کنند و کارهایشان را با کشورهای مختلف، تعامل می‌کنند. به طور کلی در شرکت‌های IT مشکلی که خیلی از شرکت‌ها دارند، یک کد را می‌نویسند ولی وقتی می‌فهمند که مناسب نیست، حاضر نیستند آن را دور بیندازند. در این حوزه یک مثالی زده می‌شود: کسب و کارهای جدید مثل برج مراقبت‌اند که فقط جریان رفت و آمد هواپیماها را مدیریت می‌کند و کاری به جزئیات ایرلاین‌ها ندارد.

## Strategy Fit

به طور کلی سه گام دارد:

(۱) عدم قطعیت‌ها از بالادست و پایین دست چگونه است؟ درک عدم قطعیت مشتریان و زنجیره‌ی

تأمین

(۲) قابلیت‌های ما در زنجیره‌ی تأمین چیست؟

(۳) رسیدن به strategic fit

### قدم اول: درک عدم قطعیت مشتریان و زنجیره‌ی تأمین

در زنجیره تأمین همیشه باید ریشه‌ی عدم قطعیت‌ها را شناخت. عدم قطعیت می‌تواند از سمت مشتری باشد و از سمت زنجیره تأمین باشد.

از نگاه مشتریان باید دید که چه سگمنتی را می‌خواهیم ارائه کنیم. هر چه قدر **طیفی** که مشتریان را تارگت کرده‌ایم گسترده‌تر باشد، عدم قطعیت بیش‌تر است. به طور مثال اگر در آموزش فقط کارشناسی ارشد و فقط دوره‌ی MBA مد نظر باشد، هر کس که بیايد بگوید من ثبت نام نکرده‌ام، سریع می‌شود به او خدمات داد ولی اگر تارگت بسیار بزرگ باشد، ابتدا از طرف پرسیده می‌شود: شما کدام مقطع را می‌گویید؟ کدام گروه؟ کدام گرایش و... در کسب و کار هم همین طور است. اگر یک سفارش در کسب‌وکار، بچه‌ها تا سالمندان را در نظر بگیرد، وقتی سفارش ایجاد شود، اول معلوم نیست مال چه کسی است؟

**کمیت** که در یک سفارش دریافت می‌شود، چه قدر است؟ (quantity of product needed in each lot) به طور مثال برای شرکت تگزاس اینسترومنت DSP محصول است که برای یک شرکت موشکی ممکن است که ده عدد سفارش شود و سفارش شرکت سامسونگ ۱۰۰۰۰۰ باشد. برای این شرکت، وقتی یک کسی می‌خواهد سفارش دهد، طیف ده تا صدهزار در ذهن شکل می‌گیرد. این طیف باعث ایجاد عدم قطعیت می‌شود.

response time customers will tolerate: به طور مثال، برای خرید از شهروند، اگر مشتری بیايد و شیر نداشته باشد، مشتری از جای دیگر خریداری می‌کند در صورتی که در بعضی از صنایع، ممکن است یک سفارش دو ماه طول بکشد. هر چه قدر زمان پاسخ، بیش‌تر باشد، عدم قطعیت بیش‌تر است. برای یک مریض در اورژانس، ممکن است که فرد سرماخوردگی داشته باشد و ممکن است که مریض، تصادفی باشد. زمان پاسخ

در نگاه **integrated** استراتژی مدیریت عملیات با استراتژی‌های زنجیره تأمین جایگزین شده است. مواردی مختلفی که در اسلاید در این قسمت وجود دارد.

عدم قطعیت می‌تواند از سمت مشتری باشد و از سمت زنجیره تأمین باشد.

برای مریضان متفاوت است و یک طیف را تشکیل می دهد. هر چه قدر این طیف گسترده تر باشد، عدم قطعیت بیش تر است.

Variety of products needed: اگر در یک فروشگاه کفش، تمامی کفش ها یک نوع باشد، می توان جعبه های کفش را منظم گذاشت در مغازه و هر بار لازم شد یک کارتن کفش را به طور تصادفی انتخاب می کنیم. این جا تنوع محصول و عدم قطعیت کم است. ولی مثلا در صنعت خودرو، شرکتی مثل بی ام و، مشتریان در اینترنت آپشن های آن را انتخاب می کنند و تعداد بی نهایت آپشن وجود دارد. این تنوع به سمت بی نهایت می رود و عدم قطعیت زیاد است.

Service Level Required: مثلا، یک مشتری ممکن است که وسط جاده ی سلفچگان هوس نوشابه ی چری کند. آیا سطح خدمات شرکت می تواند این انتظار را برآورده کند.

Price of Product: رنج محصولاتی که ارائه می شود، چه قیمتی دارد. به طور مثال شرکت های چینی from chips to ship تولید می کنند. هر چه قدر طیف قیمت های محصولات زیاد باشد، عدم قطعیت بیش تر است.

Desired rate of innovation in the product: وقتی یک مشتری زنگ می زند کدام نسل از محصول را می خواهد. 3G یا 4G؟ در نقطه ی مقابل به طور مثال، در فولاد استاندارد محصول ثابت است و می تواند تفاوت در تکنولوژی تولید باشد.

وقتی تنوع محصول بیش تر است، تقاضا برای هر محصول نامشخص تر خواهد بود و خطای پیش بینی برای هر محصول را زیاد می کند. معمولا وقتی عدم قطعیت کم باشد، حاشیه سود کم است و برعکس. مثلا، وقتی قند و شکر تولید می شود، مارجین کم است ولی حجم بالا می باشد ولی در فروش مبل، هفته ای دو دست مبل فروخته می شود ولی حاشیه سود بالا است. (forecast error)

میانگین خطای پیش بینی در عدم قطعیت بالا زیاد است. به طو رمثال در شرک اینتل حتی به هشت صد درصد هم می رسد.

در کسب و کار با عدم قطعیت بالاتر، بسیار مهم است که کالا فروش رود و انبار نشود (stock-out rate is higher).

در عدم قطعیت کم، تخفیفات آخر فصل بسیار کم تر است (forced season-end markdown).

شبیه آنچه در پایین دست (مشتریان) گفته شد، در بالا دست نیز وجود دارد:

frequent breakdown: احتمال این که تجهیزات مثل سی ان اسی، وسط عملیات به مشکل بخورند وجود دارد.

استراتژی good enough: بسیار در صنعت نرم افزار دیده می شود. به طور مثال مایکروسافت یک نرم افزار تا حدی که مناسب باشد، آن را ارائه می کند ولی ژاپنی ها ایده آلیست هستند و نرم افزار بدون نقص ارائه می دهند. بلیونر در بین آمریکایی ها بیش تر از ژاپنی ها است چون بر این استراتژی متمرکز هستند.

معمولا وقتی عدم قطعیت کم باشد، حاشیه سود کم است.

unpredictable and low yields: از بین صد محصولی که قرار بود تولید شود، حدود ۹۵ عدد آن سالم تولید می‌شود.

poor quality: حفره‌ها و کریستال‌ها در یک ورق، منجر به تغییر کیفیت یک ورق شود.

limited supply capacity: به طور مثال، در دانشکده یک ترم تقاضا برای یک درس زیاد است و همه می‌خواهند آن درس را ثبت نام کنند.

inflexible supply capacity: خیلی وقت‌ها سیستم فقط می‌تواند یک جور تولید کند. به طور مثال اگر برای تولید یک میز نیاز باشد که جایی از آن تغییر کند، امکان پذیر نیست.

Evolving production process: ابتدا که محصول وارد خط تولید می‌شود، ضایعاتش بالاست، بعدها تحت یک منحنی یادگیری، نیروی کار یاد می‌گیرد که ضایعات را کم کند.

### قدم دوم: فهمیدن زنجیره ی تأمین

آیا من می‌توانم یک عدد محصول را تأمین کنم و هزار عدد محصول را هم تأمین کنم؟

آیا می‌توانم lead time های کوتاه را پاسخ دهم؟

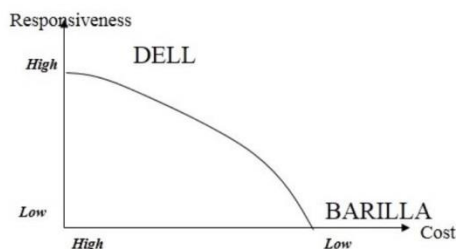
آیا می‌توانم تنوع محصولات بالا را تأمین کنم؟

آیا می‌توانم نوآوری‌های مختلفی را ارائه کنم؟

آیا می‌توانم سطح خدمات مختلف را ارائه کنم؟

آیا می‌توانم عدم قطعیت‌ها را مدیریت کنم؟ مثال، قطعه ساز ایران خودرو می‌گوید که قالب ترک برداشته است و نمی‌تواند تولید کند. اکنون ایران خودرو می‌خواهد خط تولید را متوقف کند یا این عدم قطعیت را مدیریت کند؟

وقتی می‌گوییم توانایی‌مان چه قدر است، صحبت از دو بعد زمان و هزینه می‌باشد. اگر بگویید سریع تحویل می‌دهد، هزینه‌ها بالا می‌رود و بر عکس. responsiveness و cost دو بعدی هستند که با هم در تعامل‌اند. برای هر شرکت با توجه به این دو بعد یک frontier وجود دارد.

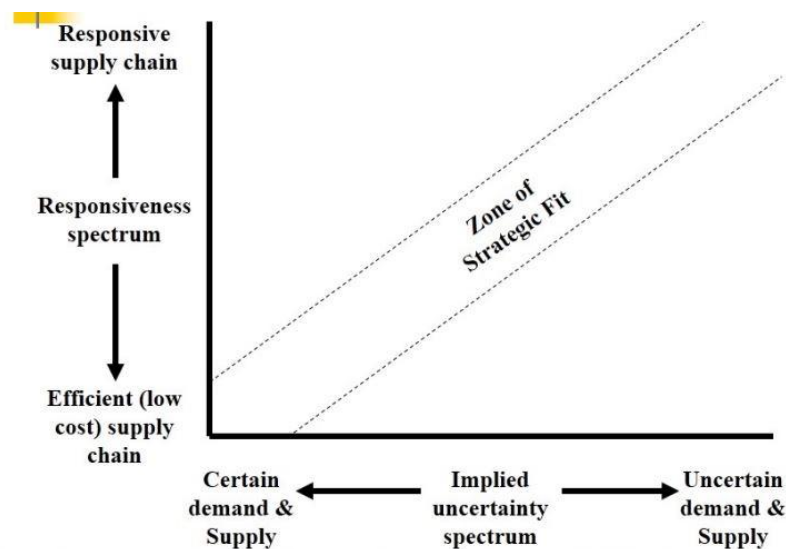


این موضوع با بعدهای دیگر هم مرتبط است.

بعضی از کسب‌وکارها، مرتبط با responsiveness می باشد. به طور مثال، موجودی کیسه‌ی خون در بیمارستان باید responsiveness قابل قبولی داشته باشند.

### قدم سوم: Achieving Strategic Fit

با توجه به دو طیف به دست آمده در دو قدم اول، یک نمودار به دست می‌آید. منطقه‌ی حول نیمساز ۴۵ درجه، منطقه‌ی strategic fit می‌باشد. یعنی اگر عدم قطعیت بالا است، responsiveness باید بالا و قابل قبول باشد. سپس با توجه به تقاضا، می‌تواند تصمیم‌گیری کرد که چگونه محصول تولید شود و به این سوال همیشگی پاسخ داد: آیا دنبال efficiency هستیم یا دنبال responsiveness هستیم؟



این موضوع با این که سیاست کلی یک شرکت ارائه‌ی محصول کم قیمت و بالاقیمت باشد، در تعارض نیست ولی حتی یک شرکتی که می‌خواهد به طور اقتصادی تولید کند باید حداقل responsive بودن را در نظر بگیرد و مثلاً یک خودروساز که خودروهای کم قیمت تولید می‌کند باید حداقل ویژگی‌ها را بر روی کالا قرار دهد. همین طور شرکتی که در حال تولید محصولات با responsive بالا می‌باشد، لازم است که حداقل efficiency را در نظر بگیرد.

**Postponement:** هر چه می‌توان فرآیند تولید تنوع محصول را به تأخیر بیندازیم، بدون این که تقاضا را دیر تأمین کنیم، به فضای postponement رسیده‌ایم. مثلاً، در صنعت خودرو، ممکن است خودرو تولید شود، و جاهایی که می‌خواهد صندلی و داشبورد قرار دهیم، از مشتری می‌پرسیم که چگونه باشد و آپشن‌ها را در اختیار مشتری می‌گذاریم. به اصطلاح differentiation را تا دقیقه‌ی نود به تأخیر می‌اندازیم. در فضای postponement باید از تقاضای محصولات از قبل آگاه بود که این کار سختی است. یک شرکت تولید لباس ایتالیایی برای رفع مشکل عدم دانستن تقاضای رنگ لباس مشتریان، لباس‌های بی‌رنگ تولید می‌کند تا هر

یک طیف وجود دارد که یک سر آن highly effecient و سر دیگر آن highly responsive می باشد.

هر شرکتی حتی با استراتژی Low Price هم می‌تواند در محدوده‌ی خودش، strategic fit را اجرا کند.

یکی از راه‌ها این است که ماژولاریتی را بالا ببریم. به سمت پلتفرم رفتن، سرعت و تنوع محصول را بالا می‌برد.

در فضای postponement در واقع، به جای این که تنوع محصول را تا انتها بیاوریم، در دقیقه‌ی نود، دست به تنوع می‌زنیم.

موقع مشتریان سفارش دادند، بتواند با توجه به تقاضای آن‌ها لباس‌هایشان را به بازار ارائه دهد. Postponement می‌تواند در هر کجای عملیات رخ دهد. به طور مثال، تولید پرینتر، نیاز به آداپتور متناسب برق ۲۲۰ ولت یا ۱۱۰ ولت دارد. یک رویکرد این است که آداپتور در مراحل اول تولید، برای پرینتر انتخاب شود و پس از تولید، محصولات به قسمت بازار خود که ۱۱۰ ولت یا ۲۲۰ ولت می‌باشد برود. رویکرد دیگر تولید پرینترهای یکسان است که پس از تولید، متناسب با مکانی که قرار است پرینتر استفاده شود، آداپتور آن انتخاب می‌شود.

وقتی می‌خواهیم قیمت‌گذاری انجام دهیم، در حالت efficient، تا حد ممکن ارزان‌تر تولید می‌کنیم.

Efficient SC	Responsive SC	
تا حد ممکن ارزان‌تر تولید کنیم.	سود بالاتری گرفته می‌شود.	قیمت‌گذاری (pricing)
بهره‌وری منابع تا حد امکان بالا باشد.	بخشی از ظرفیت را عمداً خالی گذاشته می‌شود تا برای تقاضاهای دقیقه‌ی نود، پاسخ داده شود.	تولید
حداقل موجودی مدّ نظر است.	تا جای ممکن سعی می‌شود که از محصولات جدید، موجود باشد	موجودی
تا جایی زمان تحویل را کاهش می‌دهیم، که هزینه‌ها خیلی بالا نرود.	با هر قیمتی، پاسخ‌گویی را می‌بریم بالا	Lead time
تأمین کننده‌ای که ارزان جنس را برساند دستم.	از تأمین کننده‌ای می‌خرم که سرعت و انعطاف بالایی دارد.	Supplier

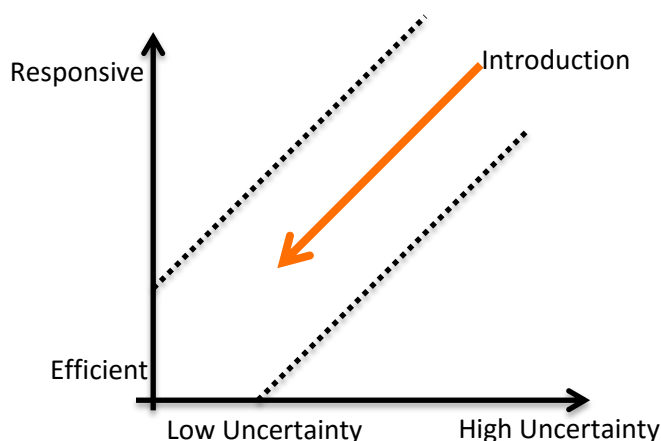
یک شرکت، می‌تواند بعضی از محصولاتش را بر اساس منطق Responsive SC و بعضی دیگر را بر اساس منطق Efficient SC ارائه دهد.

یکی از مشکلات شرکت‌ها این است که مدیر تدارکات، در تأمین مواد اولیه به صورت Efficient عمل کند ولی در کانال توزیع می‌خواهد، Responsive عمل کند که این منجر به ایجاد اختلال در عملکرد شرکت می‌شود. نباید برای تولید یک محصول بعضی از عملیات با منطق کارایی و بعضی از عملیات با منطق سرعت در پاسخ‌گویی پیش برود.

عرفا معتقداند شجاعت در عمل، فرزند یقین در نظر است. کسی که در اندیشه‌اش به یقین رسیده، در عم هم با جرئت عمل می‌کند. به طور مثال، شرکت آمازون به طور بلند مدت، ارزان قیمت‌گذاری می‌کرد تا سهم بازار را به دست آورد و در حجم بالای فروش، بتواند به سودآوری برسد و به نوعی این پاسخی به منقدان آمازون بود. مورد دیگری که به آمازون کمک کرد، استفاده از مفهوم learning curve بود که توسط مشاوران بوستون ارائه شده بود. این مفهوم می‌گوید که هر چه نرخ تولید، بالا برود، متوسط هزینه‌ها کم می‌شود. در این حالت، آمازون و شرکت‌های چینی، در یک سطح تولید، قیمت را پایین‌تر از آنچه باید باشد، ارائه می‌کنند و به طور بلند مدت، تقاضا را جذب می‌کنند.



Product Life Cycle: هر چه به سمت اوایل ارائه‌ی محصول می‌رویم، عدم قطعیت بالا است و باید یک استراتژی داشته باشیم و هر چه به سمت آخر دوره‌ی عمر محصول می‌رویم، محصول به سمت Commodity می‌رود. در نتیجه در اوایل ارائه‌ی محصول، بهتر است رویکرد responsiveness می‌باشد و هرچه از عمر آن می‌گذرد به سمت کارایی باید رفت. دوران launch محصول، دینامیک متفاوتی با دوران establishment محصول دارد و خطا است که قواعد بازی یک نهال را برای یک درخت تنومند اعمال کرد.



با توجه به حوزه‌های مختلف زنجیره‌ی تأمین، باید برای یک محصول، رویکرد consistent را اجرا کرد. اگر گام‌های مختلف زنجیره‌ی تأمین نظیر تأمین کنندگان، تولیدکنندگان و... را در نظر بگیریم، در هر گام می‌توان strategic fit را اعمال کرد. اگر می‌خواهیم efficient عمل کنیم، تمام محصولات هر کجا ارائه می‌شود باید سازگار عمل شود و همه جا یک رویکرد داشته باشیم. زمان ارائه‌ی محصول اگر در شمال یا جنوب ارائه می‌شود، یک جور باشد. این که از کجا شروع شود، بستگی به تجربه دارد. مثلاً وقتی یک شرکت تعداد زیادی مشتری و شرکت‌های زیر مجموعه دارد، این که از کجا strategic fit را شروع کند، بستگی به تجربه‌ی شرکت دارد. باید به این سمت پیش رفت که به صورت global consistence عمل کرد.

## مدیریت خدمات

هر چه زمان می‌گذرد، مرز بین محصول و خدمات کم‌رنگ می‌شود. سهم اشتغال در آمار کشورها در حال زیاد شدن است. به طور مثال در آمریکا از ۵۹ درصد سال ۱۹۶۵ به ۷۸ درصد سال ۲۰۰۵ رسیده است. این روند تنها برای کشورهای صنعتی و پیشرفته نیست، بلکه برای کشورهای در حال توسعه هم می‌باشد. برای کشورهای جهان سوم که صاحب تکنولوژی نیستند، حوزه‌ی خدمات بسیار مؤثر است. خدمات اجتناب ناپذیر است. در زمانی که درهای کشور باز باشد و تکنولوژی وارد کشور شود، برای ارائه‌ی تکنولوژی نیاز به خدمات است.

کم شدن سهم اشتغال کشاورزی، به معنای کاهش اهمیت آن نمی‌باشد. بلکه بدین معناست که در حال مکانیزه شدن می‌باشد که نیاز کمتری به نیروی انسانی دارد. در صورتی که در حوزه‌ی خدمات همیشه به نیروی انسانی نیاز است.

خیلی وقتها کالا می‌خریم  
می‌بینیم که با خدمات باندل  
شده است و خیلی وقتها  
خدمات می‌خواهیم بخریم که  
می‌بینیم با کالا باندل  
شده است.

در ترکیه، شرکت‌های نرم‌افزاری تمام ورشکست شده‌اند. چون توانایی رقابت با مایکروسافت و موارد دیگر را نداشته‌اند. در صورتی که در ایران به دلیل بسته بودن درهای کشور، شرکت‌های نرم‌افزاری موفق بوده‌اند.

یک طیفی بین کالا و خدمت وجود دارد. به طور مثال وقتی سوخت‌گیری انجام می‌شود، ابتدا خدمتی عرضه می‌شود که سوخت از پالایشگاه تا سطح شهر می‌آید و سپس کالای بنزین ارائه می‌شود. این طیف پیوسته است.



## Service Classification

چند روش برای service classification ارائه می‌شود.

یکی از روش‌ها این است که ببینیم، چه قدر یک خدمت، استاندارد است و یا میزان customization کم یا زیاد است و بعد دیگر، degree of labor intensity به معنای نسبت هزینه‌ی نیروی انسانی به هزینه‌ی capital که مصرف می‌شود. این بعد اگر high باشد، یعنی سهم نیروی انسانی نسبت به تجهیزاتی که به کار می‌رود پررنگ است و برعکس. این دو بعد یک ماتریس را ایجاد می‌کند.

Degree of interaction and customization		
	Low (standard)	High (customized)
Degree of labor intensity (labor cost/capital cost)	<u>Service Factory</u> • Airlines • Trucking • Hotels • Resorts & recreation	<u>Service Shop</u> • Hospitals • Auto Repair • Other repair services
	<u>Mass Service</u> • Retailing • Wholesaling • Schools • Retail aspects of commercial banking	<u>Professional Service</u> • Physicians • Lawyers • Accountants • Architects

## چالش‌های مدیران خدمات

با توجه به ماتریس معرفی شده، حالت‌های مختلفی داریم:

وقتی که service factory و service shop مدّ نظر باشد. نقش نیروی انسانی کم‌تر است و تصمیم‌گیری سرمایه‌گذاری خیلی مهم است. این که چه تکنولوژی خریده شود، مهم است. اگر تقاضا کم و زیاد می‌شود، با این سرمایه‌گذاری مثلا فقط می‌شود یک هواپیما خرید. منابع محدود است و تقاضا هم کم و زیاد می‌شود، باید برنامه‌ریزی کرد.

وقتی mass service و professional service مدّ نظر باشد، نیروی انسانی وزن پررنگی دارد. تمام جنبه‌های نیروی انسانی خیلی مهم است و تکنولوژی خیلی وزنی ندارد.

وقتی service factory و Mass Service مهم باشد، بازاریابی خیلی مهم است. به طور مثال، شرکت‌های ایرلانی که می‌خواهند تمایز ایجاد کنند، شاید اینترنت حین پرواز ارائه می‌دهند.

وقتی service shop و professional service باشد، هزینه‌ها زیاد بالا می‌رود. چون خدمات متنوعی ارائه می‌شود، دیر یا زود هزینه‌ها بالا می‌رود. برای مدیریت هزینه‌ها نیاز به نرم‌افزار خاص، دستیار و... وجود دارد. در واقع باید کیفیت بالایی را ارائه داد. هزینه‌ها در اینجا سریع بالا می‌رود. در این قسمت باید دغدغه‌های عملیاتی از جنس efficiency باشند. هزینه‌های ثابت، در این حالت، سریعاً بالا می‌رود.

از دیگر منظرها نیز می‌شود که service classification را بررسی کرد:

آیا با اشیاء در حال تعامل هستیم یا با افراد؟ مثلا برای تعمیر ماشین، در مکان سرد هم می‌شود انجام داد. ولی وقتی جان آدم‌ها مطرح باشد، یک نوع خدمت دیگری ارائه می‌شود. دغدغه‌ها در دو حالت متفاوت است. هر جا که حضور آدم وجود داشته باشد، شرایط محیطی مهم است. اگر سراغ بانک می‌رویم، بانک اگر همه‌ی خدمات را خوب انجام بدهد و انواع حساب‌ها را ارائه کند، نمی‌تواند که از نورپردازی محیط، جای پارک و صندلی مناسب برای مشتریان، غافل شود. یا رستوران نمی‌تواند غذای خوب و با قیمت مناسب ارائه دهد ولی در محیط رستوران مگس و سروصدا وجود داشته باشد. core service را خوب ارائه کردن کافی نیست. بعد دیگر این است که خدمت tangible می‌باشد یا خیر؟ به طور مثال، درمان و آرایشگاه tangible می‌باشد ولی آموزش، intangible است. یک نفر می‌تواند بگوید که من چیزی یاد نگرفته‌ام و کسی دیگر بگوید که من کلی چیز یاد گرفته‌ام. خیلی subjective است. کلاً خدمات intangible است. منتها یک طیفی دارد که بعضاً خیلی soft است ولی بعضی را حداقل می‌شود ادراک کرد.

یک جنبه‌ی دیگر، از دیدگاه عضویت است. مثلا بانک و بیمه از این نوع هستند که افراد نیاز به membership دارند در صورتی که رادیو این طور نمی‌باشد. از طرف دیگر، بعضی خدمات پیوسته و بعضی گسسته هستند. چیزی که به طور دائم در حال ارائه می‌باشد، مدیریت سخت‌تری دارد مثل نیروگاه. ممکن است که نیاز باشد با کشورهای همسایه در تماس باشیم که اگر نیروگاه دچار مشکل شد از نیروگاه کشورهای همسایه استفاده کرد. ولی خدماتی که discrete باشد، متفاوت است. مثلا یک تعمیرگاه در روزها و ساعات خاص خدمات ارائه

core service را خوب ارائه کردن کافی نیست.

می‌دهد و مدیریت راحت‌تری دارد. هر چه خدمت memberbased باشد، می‌توان از مصرف مشتری و رفتار مشتری آگاه بود و با توجه به آن تصمیم‌گیری کرد. مثلاً قبلاً نرم افزارها membership کمتری داشت ولی تولیدکنندگان نرم افزارها، آگاهی کمتری نسبت به مشتریان داشتند. ولی اکنون ارائه‌دهندگان نرم افزارها از محصول به سمت خدمات رفته‌اند و خدمات مصاحبات ابری ارائه می‌دهند و membership وجود دارد و رفتار مشتریان معلوم می‌شود تا بدانند در چه حوزه‌ای، چه محصولی را ارائه دهند. مثلاً توپوتا، خودرو به صورت قسطی بدون کارمزد ماشین را ارائه می‌دهد. هدف توپوتا این است که با مشتریان در تعامل باشد و از آن‌ها اطلاعات راجع به محصولات بگیرد. شرکت‌ها دوست دارند که از non-membership به سمت membership حرکت کنند. به طور مثال، گوگل میلیاردها خرج R&D می‌کند ولی خدمات موتور جستجو را رایگان ارائه می‌دهد. گوگل از داده‌های افرادی که جستجوی می‌کنند، بسیار بیش‌تر کسب درآمد می‌کند و می‌فهمد که مثلاً gap از لحاظ برندی، در چه وضعیتی قرار دارد و گوگل market insight قوی دارد و در حوزه‌های مختلفی از جمله بازار سهام و... نقش بازی کند. وقتی یک محصول به سمت خدمت می‌رود به این حالت می‌رسد که شرکت به جای گفتن میزان فروش خود، از میزان اعضای خود می‌گوید.

مورد دیگر، بعد گسترده و نازک بودن را در بر می‌گیرد. به طور مثال بیمه‌ها از نگاه steady state تقریباً ثابت هستند ولی حوزه‌های الکترونیکی و تلفن خدمات گسترده‌تر است. بعد دیگر هم، نگاه از اوج کاری است. به طور مثال سی تیرماه، زمان بررسی مالیات‌ها در ایران است و اوج تقاضا وجود دارد. در این حالت تقاضا از ظرفیت بیش‌تر می‌شود.

مورد دیگر، ارائه‌ی خدمت از یک سایت یا سایت‌های مختلف است. به طور مثال، تاثیر در یک سایت ارائه می‌شود در صورتی که یک رستوران ممکن است شعبه‌های مختلف داشته باشد. بعد دیگر این است که آیا مشتری به سمت سایت ارائه‌ی خدمات می‌رود، یا اینکه ارائه دهنده‌ی خدمت مثل تاکسی به سمت مشتری می‌رود. مدیریت توزیع از تولید بسیار سخت‌تر است، در تولید نیروها را می‌توان بازرسی کرد ولی در توزیع، نیروها می‌توانند بهانه‌های واهی مثل ترافیک بیاورند و نمی‌توان آن‌ها را validate کرد. تکنولوژی می‌تواند به مدیریت در توزیع کمک کند. حالتی وجود دارد که transaction در arm's lenght می‌باشد. بدین معنا است که وقتی خدمات را ارائه می‌کنیم، supplier کیست؟ به طور مثال، هر کسی که قواعد را رعایت کند، می‌تواند خدمت را عرضه کند. به طور مثال کارت اعتباری این طور است. مثال دیگر این است که اپراتورهای تلفن همراه مثل همراه اول و ایرانسل، زیرساخت را ارائه می‌کند و هر کس که شرایطش را داشته باشد، با توجه به زیرساخت‌های آن‌ها، خدمات ارائه می‌دهند.

## Service package

معمولاً در پکیج خدمات چهار چیز وجود دارد:

۱- Supporting facility: مثلاً رفتن به اسکی.

۲- Facilitating Goods: مثلاً برای اسکی چوب اسکی ارائه می‌شود.

شرکت‌های آمریکایی کم‌تر از شرکت‌های کره‌ای و چینی diversified تولید می‌کردند که این حوزه‌ی data آن‌ها را به این سمت برده است که diversified هم تولید کنند.

۳- خدمات صریح (Explicit service): مشخص است که چه خدمتی انجام شده است. مثلا ترمز ماشین درست می شود.

۴- خدمات ضمنی (Implicit Service): مثل این که من ماشینم دیگر تعمیر شده است و دیگر به کلاس عملیات دیر نمی رسم.

برای هر خدمت، این چهار component وجود دارد و متناسب با هر خدمت یکی از آن ها بیش تر وجود دارد. هر چه تکنولوژی توانمندتر می شود، این component ها پررنگ یا کم رنگ می شوند. به طور مثال در ebankikng بعضی از این ها کم رنگ و بعضی ها پررنگ تر شده است.

### ارزیابی خدمات

به این خاطر باید service package را در نظر گرفت چون در ارزشیابی خدمت مؤثر است. به طور مثال، اگر بانک در مکان کثیفی قرار بگیرد، دچار مشکل خواهیم شد. در محصول می توان تولید را از consumption جدا کرد ولی در خدمت این طور نمی باشد. هر چه بانک خدماتش خوب باشد، ولی کارکنان جوری برخورد کنند که انگار با یک جانی حرف می زنند، بخش explicit service می لنگد. همه ی این موارد در داوری مشتری، مؤثر است.

چیز دیگری که برای دنیای عملیات خیلی مطرح است، شرکت هایی که استراتژی low cost دارند، به این معنا نیست که شرکت از core service بزند. به طور مثال اگر استراتژی رستوران low price است، نمی تواند از گوشت گربه استفاده کند و باید core service بهترین باشد. در این حالت می تواند گران ترین گوشت را استفاده کند ولی از صندلی fancy استفاده نکند. یا مثلا ایرلاین ها با استراتژی low price نمی توانند از هواپیمای بی کیفیت و خلبان بی تجربه استفاده کنند اما می توانند به جای استفاده از فرودگاه مرکز شهر و گران تر، از فرودگاه های خارج از شهر برای کاهش قیمت ها استفاده کنند. در واقع در استراتژی low price خدمت اصلی باید بهترین باشد ولی در service package هر جا که بتوان، صرفه جویی معقول کرد، انجام می شود.

### بررسی کیس گالانز

ریشه ی اصلی خیلی از کسب و کارها attachment است. به طور مثال پروگرامرها، این مشکل را دارند. این که یکی بتواند طالب فرصت ها و حقیقت باشد، خوب است. باید نسبت به آنچه ارائه می دهیم، آزاد باشیم. شرکت گالانز، ابتدا فقط تولید می کرده است و از دیگر شرکت ها سفارش می گرفته است. (OEM بوده است). در بستر زمان، دوز و کلک های این تولید دستش می آید و گالانز مرتب سفارش می گرفت و در حجم بالا تولید می کرد. دو کاری که انجام می داد، یکی low price بودن و price war بود. دینامیک این حالت به این گونه است که وقتی قیمت کم شود، تقاضا بالا می رود و این می تواند، تقاضا را زیاد کند و پس از آن درآمد زیاد شود و بعد از ظرفیت تولید زیاد شود و دوباره باعث کاهش قیمت می شود که یک حلقه ای شکل می گیرد. (این حلقه تا یک جایی ادامه پیدا می کند). این سیاست شرکت های چینی است که حجم بالایی را تولید می کنند. ما وقتی وارد

شرکت های که استراتژی low cost دارند، به این معنا نیست که شرکت از core service بزند.

حوزه‌هایی مثل کاشی و نساجی در ایران می‌شویم اگر economies of scale نداشته باشیم، نمی‌توانیم رقابت کنیم. با توجه به این حلقه چون می‌دانیم که در نهایت مثلاً می‌خواهیم پنج میلیون تولید کنیم، قیمت‌گذاری را بر این target مشخص می‌کنیم و می‌توان استراتژی low price را پیاده کرد.

شرکت‌های خوب low price عمل می‌کنند ولی وارد جنگ قیمتی نمی‌شوند. جنگ قیمتی به این معنا است که قیمت را کم کنیم تا حال رقیب را بگیریم. شرکت‌های موفق هیچ وقت چنین کاری نمی‌کنند که مکنزی در گزارش خود، مفصل بیان می‌کند. جنگ قیمتی، روی accident مثل متلک شروع می‌شود و دامن گیر همه است و در آخر هم برنده ندارد. دلایل مختلفی وجود دارد که شرکت‌های خوب از جنگ قیمتی پرهیز می‌کنند: emotionality جایگزین rationality می‌شود. مکنزی گزارش می‌دهد که شرکت‌هایی بوده‌اند که پنج سال وام می‌گرفته‌اند تا رقیب را از بین ببرند. low price یک زیرساختاری دارد که هیچ کس این شرکت که low price را ارائه می‌کند به دید کسی که می‌خواهد من را از بین ببرد نگاه نمی‌کند.

کسی که تولید کند و لایسنس یک شرکت خارجی را داشته باشد، OBM بازار دستش می‌آید و یاد می‌گیرد که minor modification انجام دهد. نمی‌تواند تکنولوژی را تغییر دهد، چون وقتی می‌تواند این کار را انجام دهد که R&D قوی داشته باشد که در نهایت با یک OBM قوی داشته باشد. OBM کسی است که فیس مارکت جلوی صورتش است. OEM/ODM فقط فضای supply side جلویش است در صورتی که OBM بخش تقاضا جلویش است. اصل پول زنجیره تأمین دست صاحبان برند است و کسی که تولید می‌کند، حاشیه سود محدودی دارد. کسی که صاحب برند است می‌تواند R&D های جدی را انجام دهد و بازار را جلو ببرد. OBM شدن یعنی بازار را مدیریت کردن که کار ساده‌ای نیست و core competency ویژه می‌خواهد. اکثر شرکت‌های چینی با OEM شروع کردند و سپس به OBM رسیده‌اند. وقتی یک شرکت بزرگ شده است اگر بخواهد فقط OEM بماند، می‌تواند نهایتاً کل بازار چین را بگیرد در صورتی که margin اصلی دست OBM است. این شرکت چینی، سه استراتژی در این مورد داشت. یکی که OEM/ODM بماند. دیگری OBM هم را به حالت قبلی اضافه کند و حالت آخر این است که فقط OBM باشد.

کسی که از دنیای production باشد، دیر یا زود دانشش obsolete می‌شود. هم باید حداقل تکنولوژی را دانست و هم مدیران تا حدودی باید از دنیای تکنیکال فاصله بگیرند. این که فرد، در کجای طیف قرار گیرد، مهم است. اینکه اگر فرصتی دیده شد، از آن به طور کامل غافل شویم یا سریع سوئیچ کنیم، معقول نیست. باید فرصت به خوبی بررسی شود و همچنین علاقه‌ی فردی که می‌خواهد سوئیچ کند، اگر زیاد بود سوئیچ کند.

اگر کسی OEM بماند، اولین حوزه‌ای که در آن بوده است می‌تواند core competency خودش را نگه دارد ولی اشکالش این است که بازار محدود است و نهایتاً بتواند کل بازار چین را تأمین کند. در صورتی که مارجین اصلی دست OBM است. اگر وارد OBM بشود، می‌تواند وارد بازارهای دیگر هم بشود ولی باید یک سری مجموعه‌ی مهارت‌های جدید را توسعه دهد که این کار ریسک بالایی دارد. در واقع این مارجینی که اضافه می‌شود، ریسک بالایی دارد. بنابراین اگر کسی بخواهد وارد OBM شود، باید skill set را توسعه دهد و ریسک‌ها را قبول کند.

آیا فرد می‌تواند همزمان OEM/ODM و OBM هم باشد؟ تناقض این طوری است که من یک محصول با لایسنس شرکت دیگر دارم و یک محصول دارم که مال خودم است. وقتی می‌خواهم promot کنم، کدام یک را باید promot کنم؟ این OBM OEM ODM تعارض در interser ایجاد می‌کند. چه زمانی می‌توان این تعارض را کم کرد؟ وقتی که تفاوتی در یک بازار و یک شرایط ایجاد کنیم. به طور مثال گالاتز، در چین یک

در تولید بالا، ممکن است کیفیت پایین بیاید. اگر لایسنس یک شرکت دیگر را بگیرد (OBM)، می‌تواند کیفیت را بهبود دهد.

روش‌هایی برای جلوگیری از جنگ قیمتی وجود دارد. مثلاً قرارداد بلندمدت با مشتریان بسته شود.

گذار از OEM/ODM به OBM چالشی است. الزامی وجود ندارد که این گذار انجام شود.

چیز را پروموت کند و بیرون از چین، چیز دیگر را. اگر یکی بخواهد ODM/OEM را داشته باشد و هم OBM به یک ناسازگاری می‌خورد این که چگونه وارد OBM باشد و حالت های قبلی را نگه دارد که یک state of art است. بنابراین مزایای این گزینه، مثبت شدن معایب حالت قبل است. معایب مدیریت سخت این کار است. یکی از مشکلات جدی شرکت های چینی OBM است. اکثراً به خوبی تولید می‌کنند ولی برای فروش، ریششان به خارجی‌ها گیر است. گذار از OEM/ODM به OBM چالشی است. الزامی وجود ندارد که این گذار انجام شود.

	Production	R&D	Marketing	
OEM	*			
ODM	*			
OBM		*	*	

### ویژگی‌های خدمت

اول، قرار گرفتن مشتری در فرآیند تولید است. بر خلاف تولید محصول، در فرآیند ارائه‌ی خدمت مشتری هم درگیر است و به همین دلیل است که سرویس پکیج مهم است.

دوم، سرویس می‌تواند منبع innovation باشد. دلیل این امر این است که مشتری در فرآیند ارائه‌ی خدمت قرار دارد و می‌شود انواع خلاقیت‌ها را به او ارائه کرد، مثلاً مشتری بتواند مواد تولید غذایی را خودش انتخاب کند. مثال دیگر این است که در تعمیرگاه‌ها شیشه بگذاریم و مردم بنشینند و فرآیند تعمیر را ببینند. موسس مرکز چشم‌پزشکی نور، معتقد بود که بیمارستان باید مانند هتل‌ها باشد و مثلاً عمل لیزیک را بستگان بتوانند از پشت شیشه نگاه کنند.

سوم، مشتری می‌تواند نقش فعال ایفا کند. مثلاً در فست‌فودها، خود مشتری بیاید غذایی را بردارد و برود.

چهارم، استفاده از IT، مثل ebanking در خدمات زیاد دیده می‌شود.

پنجم، **Simultaneity** است. در یک رستوران، غذا را نمی‌شود انبار کرد. خدمتی که ارائه می‌شود، به طور همزمان هم مصرف می‌شود امکان decoupling وجود ندارد. کنترل کیفیت، هم مشکل‌تر می‌شود، اگر سرویس بد ارائه شود، همان موقع به دست مشتری می‌رسد و این طور نیست که مانند تولید بتوان کالای معیوب را از دیگر کالاها جدا کرد. به طور واضح‌تر این است که در تولید، موجودی وجود دارد در صورتی که در خدمت این طور نمی‌باشد.

ششم، **Pershibality** است. در ارائه‌ی خدمت، مدیریت ظرفیت بسیار مهم است. مدیریت نواسانات تقاضا با ظرفیت تأمین، کاری سخت و مهم است. در این مورد سه گزینه داریم:

۱- در سمت تقاضا، manipulation انجام دهیم. مثلاً قیمت را تغییر دهیم و بالا و پایین کنیم. به

طور مثال در روزهای غیر آخر هفته، تخفیف گذاشته شود تا تقاضا smooth شود.

۲- ظرفیت را متناسب با تقاضا، تغییر دهیم. به طور مثال، اگر استاد دانشگاه کم وجود دارد، از اساتید مدعو استفاده کنیم.

۳- نه تقاضا و نه عرضه را دست کاری کنیم. مانند یک فضای monopoly، می شود.

به طور مثال، در یک فست فود آمریکایی، افراد غذاهایی که می خواستند را چک مارک می زدند و در صندوقی می انداختند، بعد آن کاغذ اسکن می شد و متناسب با شماره، غذا برای فرد می رفت.

هفتم، **intangibility** است. در ارائه ی خدمت، انجام کار نامشهود است، پس قابل patent کردن هم نیست. ممکن است که دیگری، دقیقاً عین کار من را انجام دهد و من نتوانم کاری انجام دهم. برای حل این مشکل آمده اند و فرانچایسینگ را انجام می دهند. در واقع گفته می شود که اگر کسی می خواهد کار من را کپی کند، من خودم کمکش می کنم و از حق فرانچایس می گیریم. کسی که فرانچایس می گیرد، می تواند خدمت مشابه را انجام دهد ولی advertise کردن و... را از صاحب برند می گیرد. صاحب برند می گوید من خودم برند را تبلیغ می کنم، وبسایت می زنم، مواد اولیه ی خوب تهیه می کنم و....

هشتم، **Heterogeneity** است. در خدمات، به دلیل نامشهود بودن ادراک آدم ها از یک محصول یکسان، متفاوت است. از یک رستوران یکسان، سه نفر متفاوت، سه برداشت متفاوت دارند. IT می تواند ابزاری برای کاهش این ادراک متفاوت باشد.

### خدمت یا محصول؟

آیا تولید نرم افزار خدمت است یا محصول؟ چون می توان قبل از release کردن نرم افزار، bug هایش را پیدا کرد، پس می توان روی آن کنترل کیفیت انجام داد و در نتیجه pure service نمی باشد. وقتی که نرم افزار تولید می شود، کاملاً product است ولی در نحوه ی عرضه اش، کاملاً خدمت است. به طور مثال، اگر در موقع عرضه، وبسایت بالا نیاید، این مشکل در خدمت است.

Broadcasting، (آنچه صدا و سیما ارائه می کند) آیا خدمت است یا محصول؟ آیا مشتری در تولید فیلم نقش فعال دارد؟ در برنامه ی زنده چی؟ فیلمی که صدا و سیما تولید می کند، کاملاً product است، ولی در زمان عرضه، اگر صدا و سیما گوینده نداشته باشد، یا پارازیت یا هر چیز دیگر وجود داشته باشد، مشکل در ارائه ی خدمت است.

### Capacity Management

گاهی بر مبنای ورودی و گاهی بر مبنای خروجی ظرفیت خدمتستان، اندازه گیری می کنیم. مثلاً در بیمارستان، چند تخت خواب داریم؟ (input) یا چند بیمار در هفته درمان می کنیم؟ (output)

### Demand Management

می توان به انواع attribute های تایم اشاره کرد:

در ارائه ی خدمت، انجام کار نامشهود و غیرقابل patent کردن است.

فضای develop کردن، از جنس product است و فضای عرضه کردن از جنس خدمت است. مدیریت این دو فضا در کنار هم مهم است.



Order entry date: مثلاً، سفارشی که وارد وبسایت می‌شود.

Customer requested date: مثلاً مشتری گفته برای دو هفته‌ی دیگر سفارش را می‌خواهم.

Estimated shipment date: تخمین این که کالا تحویل داده شود، سه هفته‌ی دیگه است. (یک هفته دیر می‌رسه دست مشتری)

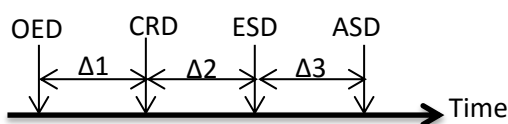
Actual shipment: زمانی که در واقعیت محصول به دست مشتری می‌رسد.

مهم  $\Delta$  این زمان‌ها است. هر کدام از این زمان‌ها یک توزیع احتمالی دارند.

$\Delta 1$ : distribution، می‌گوید که چه قدر زمان برای تحویل کالا داریم.

$\Delta 2$ : بیانگر میزان موجودی شرکت برای تحویل دادن کالا به موقع است.

$\Delta 3$ : بیانگر اتفاقات غیر محتمل است که ممکن است رخ دهد. بهتر است که به سمت صفر حرکت کند تا reliability شرکت برای تحویل کالا زیاد شود.



خیلی مواقع Dimension های مختلفی داریم. ممکنه همین آیتم را با یک نوع سفارش، از مشتری استراتژیک بگیریم یا به یک مشتری جدید بدهیم.

Bucketed nature: یعنی در یک بازه‌ی زمانی این مقدار مشخصی تقاضا وجود دارد. به طور مثال، بین ساعت ۲ تا ۳، دویست نفر به رستوران می‌آیند.

روش‌های مختلفی برای پیش‌بینی تقاضا وجود دارد که در این درس وارد آن‌ها نمی‌شویم.

وقتی در زنجیره‌ی تأمین می‌خواهیم forecasting انجام دهیم، باید unconstrained تصمیم بگیریم، بعد constraint ها را مشخص کنیم. باید موقع تنظیم objective، بدون محدودیت عمل کرد و بعد از آن، محدودیت‌ها مشخص شود، که در نهایت مجبور می‌شوید پایین‌ترش را بپذیرید، نه این‌که موقع تنظیم objective هم با اعمال هم‌زمان محدودیت‌ها عمل کنیم.

Demand shaping: تقاضا را به گونه‌ای تغییر دهیم که با ظرفیت ما متناسب شود.

در سرویس، خیلی مواقع با صف (queue) روبه‌رو هستیم. از منظر عملیاتی، یک perception queue وجود دارد و یک queue. به طور مثال، منتظر ماندن برای آماده شدن غذا در رستوران، در یک صف ایستاده خارج از رستوران و یا نشتستن در محیط جذاب رستوران، از نظر مشتری متفاوت است. وقتی که uncertainty وجود دارد، زمان سخت‌تر می‌گذرد. مثلاً وقتی گفته شود که نتایج در فلان روز و فلان ساعت اعلام می‌شود، باعث

برای مدیریت supply side باید  $\Delta$ ها را تحت بستر IT، تحت مدیریت بگیریم.

در صنعت، اکثراً از ASD برای پیش‌بینی تقاضا استفاده می‌کنند که اشتباه است.

تقاضای واقعی را CRD مشخص می‌کند. نه زمانی که شما می‌توانید کالا را تحویل دهید. در واقع می‌بایست demand side در نظر گرفته شود، نه supply side و اگر کمبودی وجود دارد، برونسپاری شود.

هر چه variability بیشتر شود، wating time هم زیاد می‌شود.

می‌شود که زمان زودتر بگذرد نسبت به زمانی که فرد برای دانستن نتایج مجبور باشد، مرتب چیزی را چک کند.

زمان بیکاری در مقابل زمان مشغولیت: وقتی منشی در حال کار باشد، زمان زودتر از حالتی که منشی بیکار نشسته‌است برای مشتری می‌گذرد.

اگر از منظر مدیریت خدمات نگاه کنیم، هر چه قدر بتوانیم، تنوع را کم کنیم، زمان انتظار کم می‌شود. به طور مثال در بانک، اگر یک کارمند فقط برای یک کار و کارمند دیگر فقط برای کار دیگر باشد، کار مناسبی نیست. در مراجعه به پزشک، اگر معاینه‌ی همه‌ی مریضان، پانزده دقیقه طول بکشد، زمان انتظار کم‌تر می‌شود تا زمانی که یک معاینه پنج دقیقه و دیگری نیم ساعت طول بکشد. مثال دیگر، وقتی درس‌ها همه سه واحدی باشد، تعداد ورودی‌های بهمن و مهر یکسان باشد، variety کم‌تر است.

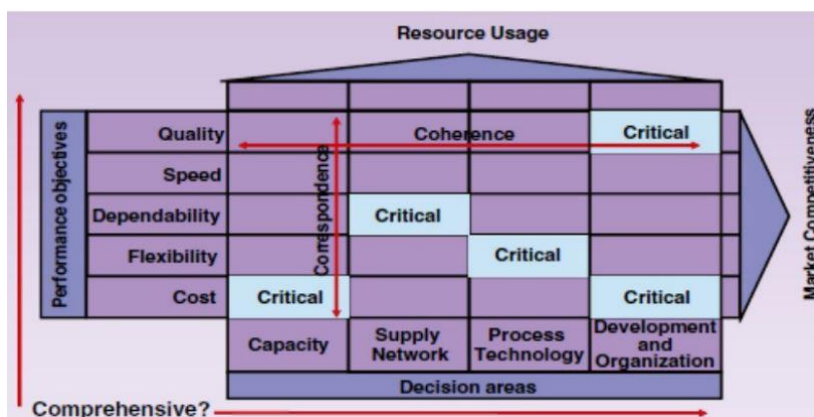
## performance measurement

کمیت‌های آن نظیر قیمت، کیفیت و سرعت و انعطاف پذیری که قبلاً گفته شد، برای خدمات هم وجود دارد.

شاخص‌های که در خدمات تعریف می‌شوند، شامل سه سطح استراتژیک strategic، tactical و operational می‌باشد.

ابزارهای ارزیابی دو نوع financial و operational دارند. هر چه قدر به سمت استراتژیک حرکت کنیم، به سمت financial می‌رویم و هر چه قدر به سمت مواردی نظیر زمان‌های انتظار و ... برویم، به سمت operational حرکت می‌کنیم.

شاخص‌های پنج‌گانه‌ای که مطرح است. اگر dependability مهم است، باید به supply network بیش‌تر توجه کرد. معمولاً در استراتژی سازمان، تمام پنج مورد مهم نیست، ولی این می‌تواند به عنوان یک نقشه باشد که سازمان بفهمد که به کدام بخش باید، بیش‌تر توجه کند. وقتی گفته می‌شود که کدام critical است بدین معنا نیست که باید از موارد دیگر صرف نظر کرد.



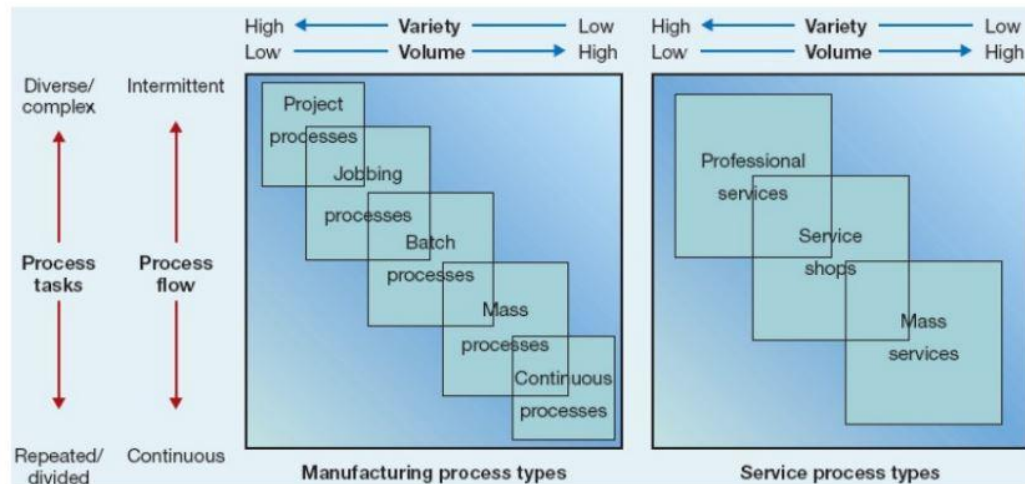
روند جهانی این طور است که در شرکت‌ها فضای محصول (product) کم می‌شود و حوزه‌ی خدمات آن زیاد می‌شود. این هنر است که ما به جای فروختن spot بتوانیم steady cash flow داشته باشیم. در این حالت می‌توان برنامه‌ریزی کرد. این به state of art هم بر می‌گردد که چه طور وارد حوزه‌ی خدمات شویم. مثال برای این مورد، مثال نرم‌افزار و محاسبات ابری است. در این حالت می‌دانیم که فلان شرکت هر سال به خاطر این خدمت، این قدر پول به ما پرداخت خواهد کرد که می‌توان روی این برنامه‌ریزی کرد. یک بیمارستان مثلا می‌تواند یک spot market باشد. یک موقع دیگر هست که بیمارستان دنبال قرارداد بستن با سازمان‌ها است مثلا به شرکت‌ها می‌گوید که من با شما قرارداد می‌بندم تا سالیانه کارمندان را چک آپ کنم. در این حالت، سرویس ارائه می‌دهد. شرکت‌ها به این خاطر به سمت خدمات می‌روند که می‌توان روی آن حساب باز کرد.

مثال IBM: این شرکت در زمانی دچار بحران شد و سود آن به شدت افت کرد. به این خاطر که نرم‌افزار و سخت افزار compatible اعلام شد و دولت به او ایراد وارد کرد که تو داری حالت monopoly انجام می‌دهی. در این حالت مدیرعامل آن به این نتیجه رسید که IBM دیگر نباید محصول بفروشد و Golobal Solution Delivery Division را ارائه کرد. IBM مبدع ارائه‌ی راه حل شد. در حالتی که اکثر شرکت‌ها تخصصی کار می‌کردند، IBM گفت که از الف تا ی را در حوزه‌ی IT ارائه می‌دهم و مبدع total solution شد. در حالتی که total solution نباشد، سخت‌افزارها مشکل را بر سر نرم‌افزارها می‌گذارند و برعکس. در نهایت، IBM با راه‌حلی که ارائه داد، دیگر دغدغه‌ی نرم افزار یا سخت افزار مطرح نبود. نگاه service oriented برای داشتن steady cash flow نیاز است. حتی بعضی از ایده‌های رادیکال، اجازه دادن تابلوهای زیبایی را می‌توان نوعی از خدمات دانست.

#### مقایسه‌ی خدمات و تولید از لحاظ فرآیند

گفته می‌شود که هر چه حجم بیشتر باشد، تنوع کم می‌شود. در حوزه‌ی فرآیندهای سرویس یک قُطر مطرح است که از حوزه‌ی خدمات گسترده به سمت خدمات حرفه‌ای حرکت می‌کند. در حوزه‌ی تولید هم، یک قُطر مطرح است. اگر هر دو با هم ترکیب شود، به یک قطر feasible می‌رسیم. این که به کدام سمت برویم، بستگی دارد که چه قدر فرآیندهایمان flexible و.... است.

نگاه service oriented برای داشتن steady cash flow نیاز است.



## layout

چگونه چیدمان فعالیت‌ها را طراحی می‌کنیم؟

### Fixed layout

ما خودمان جابه‌جا می‌شویم و آیتم‌ها این قدر بزرگ‌اند که نمی‌توان آن‌ها را تغییر داد. مثل کشتی سازی و هواپیماسازی. یا مثلاً وقتی می‌خواهیم در بیمارستان یک عمل جراحی انجام شود، همه‌ی متخصصین در اتاق عمل جمع می‌شوند در صورتی که هر کدام ممکن است جداگانه دفتر داشته باشند. در رستوران، یک موقع هست که خود فرد می‌رود غذا می‌کشد و به موقع هست که فرد خودش انتخاب می‌کند و غذا را می‌آورد.

### Functional layout

آنچه که از یک جنس باشد را کنار هم قرار می‌گذاریم. می‌خواهیم یک چیز تولید کنیم، تمام جوشکاری را به دیپارتمان جوشکاری می‌دهیم. برای رنگ و .... هم همین طور. مثال در حوزه‌ی خدمات: در شهرداری، دیپارتمان نقشه برداری طبقه‌ی چهارم، جواز گرفتن طبقه‌ی سوم و....

### Celluar layout

این گونه هست که بر اساس نوع خروجی کار، تمام کارهای مختلفی را که به خروجی مربوط است را کنار هم می‌گذاریم. به طور مثال در بیمارستان، یک بخش را فقط برای کودکان اختصاص می‌دهند. هر آنچه که نیاز داشته باشد از جمله عکس گرفتن و.... کنار هم قرار دارد.

### Product layout

درست بر عکس fix است. محصول حرکت می‌کند و بر روی آن کارهای مختلف انجام می‌شود.

هر چیزی که سرویس دریافت می‌کند اگر حجیم باشد، ثابت می‌شود.

خیلی وقت‌ها functional و cellular می‌تواند ترکیب شوند. به طور مثال، در یک فروشگاه چندین طبقه وجود دارد که هر کدام مربوط مثلاً به خانم‌ها و دیگری مربوط به آقایان است. در بخش مربوط به آقایان، بر اساس function آن‌ها، طبقه‌بندی می‌شود. مثلاً اسپرت و... در شهرداری هم می‌تواند این طور باشد که چندین طبقه باشد که مثلاً یکی برای باغ و دیگری برای ساختمان. سپس هر طبقه، کلیه‌ی خدمات نظیر نقشه‌برداری و دادن جواز را انجام دهد.

انواع layout با فضای سرویس و تولید گفته شده، مرتبط است. وقتی خدمات حرفه‌ای باشد و یا تولید، project process باشد، layout ثابت مناسب است.

اگر حجم را بر اساس هزینه برای layout های مختلف در نظر بگیریم، می‌توان بر اساس حجم تصمیم‌گیری کرد. وقتی که حجم بالا باشد، استفاده از product layout به صرفه می‌باشد و وقتی که حجم کم باشد، استفاده از fix layout به صرفه‌تر است.

## (Purchasing) Material Management

مدیریت materail بیش‌تر در حوزه‌ی صنایع پایین دستی مطرح است ولی در صنایع بالادستی هم مطرح است. کدام نوع از جریان مطرح است؟ اطلاعات، Material، funds. که این‌جا material مطرح است. مواد باز یافتی، مثالی از این است که مواد از پایین دست به بالا دست برود یا کالاهای مرجوعی هم همین‌طور است. معمولاً حرکت مواد از بالادست به پایین دست است. پول از پایین به بالا می‌رود و برای کالاهای مرجوعی، از بالا به پایین است. جریان اطلاعات از هر دو طرف است.

دو نوع مواد در صنایع وجود دارد:

direct material: همه چیزهایی است که در خروجی مصرف می‌شود. به طور مثال، در تولید خودرو، ورق و چرم استفاده می‌شود.

indirect material: به طور مثال، در ایران خودرو، نیاز به کامپیوتر و لوازم تحریر وجود دارد. این‌ها به طور مستقیم، در خروجی دیده نمی‌شوند.

مواد غیر مستقیم، small order size است، مثلاً به تعداد نیاز کامپیوتر خریده می‌شود. در صورتی که direct material، large order size می‌باشند. بر خلاف آنچه تصور می‌شود indirect material ها بسیار

functional به جورایی مستقل از خروجی است و فقط کارهای مشابه را کنار هم می‌گذارد.

به طور مثال، وقتی میز و کتابخانه ساخته می‌شود. ممکن است که هر کدام نیاز به جوشکاری داشته باشد، هر کدام ممکن است که بیایند به بخش جوشکاری ولی در cellular خروجی مهم است و در واقع به محصول نزدیک‌تر می‌شویم.

معمولاً حرکت مواد از بالادست به پایین دست است.

مهم‌اند. شرکت‌های بزرگ مثل ایران‌خودرو و مس سرچشمه، این قدر ریخته پاش دارند که هر کس بتواند به آن‌ها راه حل ارائه دهد، می‌تواند رقم بسیار قابل ملاحظه‌ای را درآمد کسب کند.

در جریان direct material، سه بحث purchasing، warehousing و inventory management مطرح است. در این‌جا تمرکز بر روی purchasing است.

چرا purchasing مهم است؟ چون cost center است. بر روی quality هم مؤثر است.

به خاطر globalization، شفافیت اطلاعات اکثراً همه به سمت برونسپاری می‌روند. به خاطر این فضا، نیاز به تأمین زیاد شده است. در جهان جدید بر این باوریم که نیاز نیست همه چیز را خودمان داشته باشیم و خیلی از چیزها را باید از بیرون بخریم و نیاز به خرید خیلی بالا رفته است.

مثال: یک خودروساز، تمام توجه به سمت پایین دست است. کسی برنده است که سیگنال بازار را بهتر بفهمد، وقتی بفهمد که مارکت SUV می‌خواهد، همه چیز را می‌رود می‌خرد و SUV را ارئه می‌کند. کسی برنده است که بداند در پایین دست، market به چه چیزی نیاز دارد.

$$\text{Net Profit} = \text{Price} - \text{COGS}$$

$$\text{COGS} = \text{Purchasing} + \text{Labor} + \text{Resourcing}$$

در کاهش هزینه، purchasing یکی از پارامترهای کلیدی است. به طور مثال در صنعت کاغذ، purchasing پنجاه درصد COGS می‌باشد.

حوزه‌ی purchasing، به شدت سیاسی است. به طور مثال، وقتی درصد خرید از ۴۰ درصد به ۴۵ درصد برسد، ممکن است تفاوت میلیاردها باشد، به همین جهت، برای این میلیاردها، موارد سیاسی زیادی رخ دهد. در این خریدها که اعداد نجومی است حتی در خارج از کشور هم، فساد زیادی دیده می‌شود.

در حوزه‌ی جاسوسی، این مسائل دیده می‌شود. فرض کنید روسیه جاسوس می‌فرستد اروپا و آمریکا و آن‌ها هم می‌فرستند. معمولاً این جاسوس‌ها دو طرفه در می‌آیند و این سخت است که بفهمند که جاسوس خودی اطلاعات را لو داده است. برای فهمیدن این اطلاعات، به agent‌های مختلفشان، اطلاعات متفاوت می‌دهند و وقتی اطلاعات لو برود، معلوم می‌شود که کدام agent لو داده است. مأمورها معمولاً با هم، ارتباط ندارند.

### بررسی کیس بیمارستان شولدیس

این بیمارستان درآمدزایی خوبی داشت و بیش‌تر به هتل شباهت داشت تا به بیمارستان. فضای هتل مانند، از جنبه‌های سرویس پکیج است. فضا کاملاً ریلکس بوده است و مریض و دکتر و صاحب بیمارستان و سرمایه‌گذار، خوشحال‌اند و این به خاطر عملیات درست است. اگر عملیات خوب، باشد همه خوشحال‌اند و اگر عملیات نامناسب باشد، همه در عذاب‌اند. ظرافت‌های عملیاتی این است که فایلشان را از قبل پر کنند به جای این که

در بحث purchasing  
بیش‌تر، تمرکز بر روی direct  
material می‌باشد.

برونسپاری بیش‌تر، به معنای  
نقش پررنگ‌تر purchasing  
است.

سوال این است که شما در  
مقام مدیرعامل، چگونه فضای  
خرید را بنا کنید تا از فساد  
جلوگیری کنید؟

بیایند در بیمارستان و در فضای بیمارستان که استرس دارد، پُر کنند. هر چه خارج از بیمارستان، Off load شود، بهتر است. target market که این جا انتخاب شده است، پیامش economic of scope است. همه را پاسخ گو نبود، بلکه کسانی که از یک سلامت نسبی برخوردار بودند، هدف آن ها بوده اند. variety کم بود تا صف طولانی نداشته باشد. کنترل کردن دامنه ی variety خودش، نوعی state of art است. در این فضا دلهره و استرس وجود دارد، این که مریض های قبلی بیایند، دلداری بدهند، خیلی کمک می کند. این که مریض قبلی بیاید و بگوید من دیروز عمل کردم، الآن وضع خوبی دارم. layout هم به خوبی طراحی شده بود و مثلا تلفن در اتاق نداشتند تا فرد مجبور شود کمی راه رود و اوضاعش بهتر شود. تلویزیون را عمدا در قسمت عمومی گذاشته بودند تا socialization رخ دهد.

به لحاظ استراتژیک، سه استراتژی برایش مطرح بوده است:

افزایش تعداد طبقات و تعداد تخت

کار در روزهای تعطیل

زدن شعبه های مشابه

هر کدام از این آپشن ها را می شود، مصاحبه کمی انجام داد.

اگر خرید درست انجام شود، می توان economies of scale داشته باشیم، هزینه کم شود و در نهایت ریسک ها کم تر شود.

خرید به چه معنا است؟ ابعاد مختلف دارد.

باید کالای مورد نیاز درست را خرید. به طور مثال شرکت پرتو که توربین های مپنا را می ساخت، با این چالش روبه رو بود، چون به خاطر تحریم ها چند دست می چرخید و محصول هم تکنولوژی بالا دارد. بعضا این قدر، مشخصات فنی دارد، که انتخاب کالای درست خیلی سخت می شود. نکته ی دیگر این است که در قیمت مناسب خرید شود. معمولا برای مدیران، وقتی گفته می شود خرید، این گزینه را در نظر می گیرند.

در تعداد مناسب خریداری شود. مثلا برای یک خودرو نمی شود، سه لاستیک خرید.

کیفیت مناسب

source مناسب

زمان مناسب

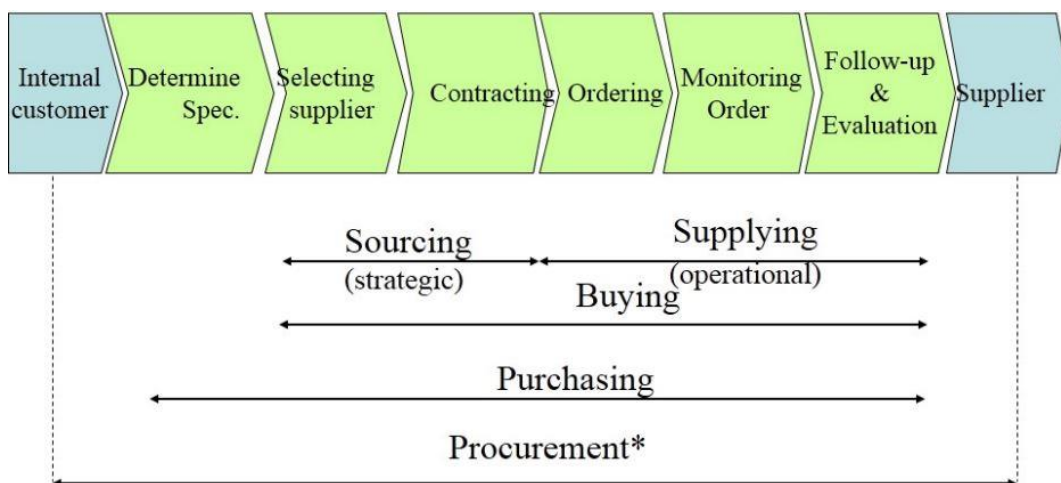
پس در تعریف purchasing تمام فعالیت هایی که برای رسیدن به تمام objective های گفته شده، شکل می گیرد، مد نظر است که فقط یکی از این ها، قیمت است.

خوب است که total cost of ownership در نظر بگیریم که همه ی موارد گفته شده را در نظر بگیرد. باید در نظر گرفت که ذهنیت مدیران، بر خرید غلبه نکند.

وقتی تأمین این قدر گسترده شود، ممکن است یک شرکت جدا ثبت شود. مثل سابکو برای ایران خودرو و سازه گستر برای سایا

### تفاوت بین Supplying, Sourcing, Buying, Procurement, purchasing

از internal customer به سمت supplier کل فرآیند مشخص است. وقتی گفته می شود، procurement کل این فرآیند را در نظر گرفته می شود. برای purchasing معلوم است که چه حوزه هایی در نظر گرفته می شود. buying قسمت spec را در نظر نمی گیرد و شامل دو بخش sourcing که استراتژیک و supplying که عملیاتی است، می باشد. هر کدام از این بلاک ها شامل بخش های مختلف است.



در قسمت supplier دانستن استانداردها خیلی مهم است، کسانی که این موارد را به خوبی بلد باشند می توانند به خوبی سیوینگ انجام دهند. به طور مثال، آمدن یک کالا از منطقه ی محروم، می تواند ذخیره ی مالی در بر داشته باشد. مثلاً خریدن یک سی ان سی برای شرکت، خیلی دوز و کلک لازم دارد.

وقتی صحبت از purchasing می شود، یک مسئله ی چندبعدی است. حداقل چهار بعد را در نظر می گیریم:

بعد تکنیکال: استانداردها، کیفیت، معیارهای تأمین کننده

بعد commercial: مثلاً چه قدر درخواست داده شود؟ چه پروپزالی قبول شود.

بعد administration: انبار کجاست؟ اینویس صادر شود یا نه؟

بعد logistic: چگونگی دریافت؟ در چه سطح از موجودی سفارش داده شود؟

#### پارامترهای اثر گذار

در حوزه ی تأمین، چگونه باید از منظر organization بررسی کرد؟ (Affecting parameters)

بعد complexity: چه قدر محصول یا خدمت استاندارد است یا سفارشی شده است؟ یه موقع هست که یک محصول وجود داشته باشد ولی وقتی سوئیچ می کنیم، سطح پیچیدگی بالا می رود. اگر نصب، پیچیده باشد، پیچدگی کلی بالا می رود.

در value chain، بیش تر جنبه های مالی مد نظر است. این که هر مرحله چه مارجینی دارد و...، جنبه ی financial فعالیت ها مد نظر است.



بعد commercial uncertainty: می‌تواند کم یا زیاد باشد. یک موقع سازمان برای یک خرید جدید، باید قابلیت‌های جدید داشته باشد. مثلاً کاله برای ورود به عراق، باید یک واحد جدا برای تطبیق با فضای عراق داشته باشد. سوئیچ کردن نسل قبلی سیم کارت، می‌تواند عدم قطعیت را زیاد کند. به طور کلی منظور از commercial impact مالی است. به سادگی هر چه سائز مالی، بیش‌تر باشد، می‌توان گفت که uncertainty و دل‌نگرانی‌های بیش‌تری دارد.

ترکیب این دو بعد، یک کوآدرانت تشکیل می‌دهد. وقتی هر دو بعد کم باشد، همان تیم تأمین می‌خرد. ولی اگر از لحاظ بعد تکنیکال، پیچیده باشد، برای خرید نیازمند مهندسان هستیم. اگر بار مالی زیاد باشد ولی جنبه‌ی فنی پیچیدگی کم باشد، برای خرید بخش مالی هم باید باشد. اگر هم به لحاظ فنی و هم به لحاظ فنی، پیچیدگی وجود داشته، یک تیم cross-functional باید تشکیل شود.

	Low Product Complexity	High Product Complexity
Low Commercial Uncertainty	Purchasing	Engineering
High Commercial Uncertainty	Finance / Admin	Cross-function

مثال مینا: گفته می‌شود که انواع پیچیدگی‌های تکنیکال چگونه می‌تواند باشد؟ سه قسمت: محصولاتی که از لحاظ technicality به سه دسته‌ی low medium high تقسیم می‌شوند و برای هر کدام، یک تیم خرید تشکیل می‌شود. در low یک تیم ساده مهندسی تشکیل می‌شود، که باعث می‌شود پاسخ‌گویی زیاد باشد و در نتیجه کیفیت بالا می‌رود. در medium یک تیم joint با بخش فنی تشکیل می‌شود. و در حالت high کلاً، این مسئولیت به بخش فنی ارائه می‌شود. برای این که دانش افراد در هر بخش، به روز بماند در این بخش job rotation رخ می‌دهد. اگر محصول پیچیده باشد، تیم مهندسی فروش تشکیل می‌شود. در صورت کم‌تر پیچیده بودن، تیم فروش جلو می‌رود و تیم فنی پشتیبانی می‌کند.

تیم مهندسی خیلی وقت‌ها به تیم تولید کاری ندارد و برعکس. این می‌تواند مشکل ایجاد کند. در ایران، در واقع تیم R&D وجود ندارد. برای کسانی که می‌خواهند سختی نکشد، می‌گویند بیا باید وارد R&D شوید. R&D را کسی انجام می‌دهد که بلندمدت فکر کند. R&D یعنی فکر کردن به نسل بعدی محصول. مثلاً برای موبایل، نسل پنجم می‌تواند R&D باشد.

هر کس از دنیای exutation دور باشد، دانشش obsolete می‌شود و job rotation لازم است.

**پارامترهای اثرگذار (از دید خارجی)**

replensihment lead-time: چه قدر طول می‌کشد تا یک محصول به دست ما برسد؟ بعضی از محصولات این طور است که الآن که سفارش دهید، تازه یک سال و نیم دیگر، به مرحله‌ی تولید می‌رسد.

on-time performance

supply flexibility: اگر تعداد سفارش تغییر کند یا آپشنی به محصول اضافه شود، چه پاسخ‌گویی دارد؟

Delivery frequency/minimum lot size

Supply quality

In-bound transportation cost: چه قدر از یک بندر یا حتی انبار تا مکان تولید هزینه و زمان لازم دارد؟

Pricing terms

Information coordination capability

Design collaboration capability: اگر یک قابلیت جدید اضافه شود، چه قدر می‌توان وارد بازار کرد؟

exchange rates, taxes, and duties

Supplier viability (sales voluem, profitability, debt, liquidity, etc)

supplier evaluation ، همه‌ی این پارامترها در نظر گرفته می‌شود. ممکن است که یک پارامتر دور زده شود که در نهایت منجر به یک هزینه‌ی زیاده‌تر می‌شود.

**پارامترهای موثر (داخلی)**

طبقه‌بندی ABC انجام می‌شود. مثلاً Aها، کم‌ترین تعداد و بیش‌ترین تأثیر را دارند.

جنبه‌های استراتژیک Strategic Issues :

اولین سوال اساسی این است که خودم تولید کنم یا بخرم؟ Make VS. Buy

اگر درست تشخیص داده نشود، به یک شرکت و یک کشور آسیب می‌رساند؟

اگر تصمیم بر خرید شد، از یک تأمین کننده خریداری شود یا از چند تأمین کننده خریداری شود؟

آیا قابلیت design collaboration وجود دارد یا نه؟

نگاه local vs. global view

نوع قراردادها چگونه باشد (نه به لحاظ حقوقی، بلکه به لحاظ عملیاتی)

فرآیند centralized و decentralized

بررسی Supplier Relationship Management (SRM)

**ساخت یا خرید:**

کسی که خرید می‌کند، همیشه می‌تواند بهترین تکنولوژی را با بهترین قیمت خریداری کند. ولی کسی که خودش می‌سازد، دارد یک سرمایه‌گذاری انجام می‌دهد که انعطاف‌پذیری را کم می‌کند. به طور مثال کسی که خانه خریده است، به راحتی نمی‌تواند خانه را تغییر دهد، در صورتی که کسی که مستأجر است قابلیت مانور

دارد و می‌تواند جابه‌جا کند. همیشه وقتی شما اجاره می‌کنید، می‌توانید بهترین تکنولوژی را داشته باشید ولی وقتی خودتون تولید می‌کنید، قابلیت مانور را از دست می‌دهید. بیش‌ترین قطعه‌سازانی که در صنعت خودرو مشکل دارند، شرکت‌هایی هستند که خودروساز، سهام آن‌ها را خریده است. ممکن است در خرید شرکت‌ها، شرکت‌های کوچک‌تر (که به روز بوده‌اند)، بودجه‌ی کافی برای R&D دریافت نکنند (چون مثلاً برای خودروسازی، اولویت خودروسازی است نه باطری‌سازی) و شرکت کوچک‌تر obsolete شود. اگر وارد حوزه‌ی شویم، که برای یک شرکت efficiency و core competency آن را تحت تأثیر قرار دهد، هزینه ایجاد می‌شود.

چه وقتی‌هایی باید ساخت انجام داد:

وقتی نمی‌صرفد که از بیرون بخریم

وقتی به تأمین کننده اعتماد نیست

وقتی که حاشیه سود با خرید، شدیداً کاهش یابد

وقتی وارد شدن به آن حوزه، حیاتی باشد

اگر چیزی core competency نباشد، حتی اگر تیم فنی قوی هم داشته باشیم (تیم قوی شرط کافی نیست)، بهتر است که ساخت برای آن انجام نشود.

در خارج از کشور، در صنعت خودرو، خیلی از قطعه سازها حتی از خودروسازها بزرگتر شده‌اند.

اساس ورود به پرسش make or buy، بر اساس core competency است و این‌که باید total cost of ownership را در نظر گرفت، نه فقط این‌که هزینه‌ی تولیدمان در داخل با خارج مقایسه کنیم. به طور مثال، برای هپکو، به دلیل عدم وجود proximity نمی‌ارزد که از چین غلطک بیاورد.

تشخیص core competency و قابلیت سازمانی، دشوار است. شناختن توانایی‌های شرکت مثل شناخت قابلیت‌ها و توانایی‌های انسانی است، سخت است و مربوط به مباحث استراتژی است.

طبق اصل داروین، چابکی و responsive بودن مزیت و عامل ماندگاری است، نه داشتن Asset و... زیاد، نه داشتن پول زیاد، بزرگ و... بودن، دیگر مزیت نیست. core competency‌ها، بیش‌تر intangible هستند تا tangible. مدیریت فرآیندها، مدیریت ترافیک و...، هیچ‌جا نمود بیرونی ندارد ولی intangible‌ها هستند که core competency می‌باشند.

Make: تولید معمولاً فرآیند غیرقابل برگشت (undo) است. بعد از سرمایه‌گذاری‌های انجام شده، معمولاً برگشتن به عقب سخت است و معمولاً attachment پدید می‌آید. ساختن، معمولاً، جلوی انعطاف را می‌گیرد، دیگر نمی‌توان آخرین تکنولوژی و بهترین قیمت را در زمان دلخواه، خریداری کرد. وقتی فرد خودش تولید می‌کند برای ماندگاری باید هزینه‌های R&D را پرداخت کند و گرنه رقابتی بودن و efficiency را بعد از دو تا سه سال، از دست می‌دهد. وقتی یک شرکت خودش تولید می‌کند، محدودیت در تولید و economic of

به جز core competency، همه چیز را برونسپاری کنید.

scale به وجود می‌آید. علت تمرکز هر کس در حوزه‌ی مد نظر خودش، economic of scale است. از این منظر make می‌تواند، ریسک داشته باشد.

اگر شرکتی بخواهد، specialist باقی بماند، باید بتواند بدون conflict of interest با همه‌ی شرکت‌ها کار بکند. وقتی یک شرکت سهام مدیریتی‌اش را به شرکت دیگری بفروشد، ممکن است تضاد منفعتی ایجاد شود، که این ممکن است به خاطر رقابت باشد. از طرفی پولی که در آن شرکت می‌آید، دیگر همه‌اش در آن شرکت اجازه‌ی خرج شدن ندارد، باید پول و درآمد را به parent company داد.

بعضی وقت‌ها شرکت‌هایی مثل تویوتا، supplier development انجام می‌دهند. چه طور می‌شود که supplier development انجام داد و خودم بیش از رقبا سود کنم؟

### استراتژی برای buy، make یا hybrid

تقاضا را به صورت پنج ساله در نظر می‌گیرند و NPV را برای هر حالت به دست می‌آورند. اگر به این نتیجه رسیدند که باید خرید انجام بگیرد، سوال استراتژیک بعدی این است که آیا باید از یک خریدار خریداری شود یا از چند خریدار؟ اگر خرید از یک نفر شکل گیرد، ارتباطات استراتژیک با او شکل می‌گیرد. استانداردهایمان را به او می‌گوییم و او رعایت می‌کند، حتی برای ما در plant خودش سرمایه‌گذاری هم می‌کند. در صورتی که داشتن چندین تأمین کننده عکس این موضوع است و قدرت چانه‌زنی خریدار بیشتر است. در حالتی که با چندین تأمین کننده کار می‌کنیم دیگر ریشمان گروهی کسی نیست و ریسک کاهش پیدا می‌کند. در عالم واقع هم، عمدتاً تک تأمین کننده‌ای می‌شوند، چون partnership واقعی عمدتاً با یکی ایجاد می‌شود.

در مواقعی که transaction cost کم است، یعنی ارزیابی یک تأمین کننده، هزینه‌ی کمی دارد، راحت‌تر می‌شود که بین تأمین کنندگان switch کرد. در صورتی که در زمان‌هایی که transaction cost بالا است (مثلاً باید چندین سال با تأمین کننده کار کرد، تا بفهمیم چگونه است)، switch کردن سخت می‌شود و میل به ایجاد روابط استراتژیک، شکل می‌گیرد.

وقتی transaction cost بالا می‌رود، تشکیل firm، تشکیل اتحاد استراتژیک به صرفه‌تر از خرید در بازار است، بعد که firm بزرگ‌تر و بزرگ‌تر شد، شرکت بهتر است به شرکت‌های subsidiary خرد شود و بعد به تدریج تا حد network ریز می‌شود. شبکه‌ای گسترده از nodهای فراوان، کوچک در حد یک office، این طوری efficiency بالاتر می‌رود. لانچ کردن network کار بسیار هنرمندانه‌ای است و نیاز به ظرافت‌های زیادی دارد. خیلی از وقت‌ها multisupplier در ذهن مدیران ترجیح داده می‌شود که این‌ها backup هم هستند.

رویکرد tailored sourcing: برای هر قسمت یک تأمین کننده‌ی مخصوص داریم. به طور مثال ممکن است که یک تأمین کننده داشته باشیم که تمرکزش بر روی cost باشد و به انعطاف‌پذیری پاسخگو نباشد و تأمین کننده‌ی دیگر در بخش دیگر، flexible باشد. در این حالت، برای uncertain demand، از تأمین

سایپا، معمولاً دو و بیش از دو، تأمین کننده داشت، در صورتی که ایران‌خودرو اکثراً با یک تأمین کننده کار می‌کرد.

در ابتدا که هزینه‌های مبادله در بازار زیاد است، بهتر است firm بزنیم، بعد که مجدد هزینه‌های داخلی از هزینه‌های مبادله بیشتر شد، تشکیل زیرمجموعه باعث کاهش هزینه‌ها می‌شود.

کننده‌ی flexible استفاده کنیم. بر خلاف رویکرد قبلی که backup supplierها، هم‌دیگر بودند و قدرت چانه‌زنی وجود داشت، در این رویکرد، به اقتضاء شرایط، درصد خرید از هر تأمین‌کننده، تغییر داده می‌شود. می‌شود این دو رویکرد را با هم ترکیب کرد. پس در رویکرد tailored، دنبال strategic fit هستیم.

در tailored supplying، بستگی به این که بیزینس در شروع کسب‌وکار می‌باشد یا به اندازه کافی بزرگ شده است، سهم تأمین‌کننده‌ی efficient متغیر است.

## Design Collaboration

به خاطر کوتاه شدن چرخه عمر محصولات و تنوع محصولات این موضوع مطرح است. به نوعی می‌گویند که هشتاد درصد مسائل مرتبط با purchasing مرتبط با design است. قابلیت design collaboration در این موضوع می‌تواند کمک کند. مثلاً اگر ایران خودرو یک محصول را طراحی کند و به طور سنتی هر قسمت این محصول را به تأمین‌کنندگان بفرستد، این قابلیت وجود ندارد. در صورتی که امروزه در بستر IT، در طراحی یک محصول، ممکن است که یک طراحی بیست بار revision بخورد. با داشتن این قابلیت می‌توان بیست درصد هزینه‌ها را کاهش داد، سی درصد کیفیت را افزایش داد و پنجاه درصد زمان بازار (time to market) را کاهش داد که مورد آخر بسیار مهم است.

در بحث purchasing ممکن است که مثلاً یک قطعه‌ساز، مواد اولیه را از یک شرکت بخرد و بر روی آن عملیات ساخت را انجام دهد و سپس بفروشد و در حالت دیگر این باشد که یک شرکتی که قطعه را لازم دارد، خود مواد اولیه را بخرد. شرکتی که قطعات را لازم دارد، در صورتی این کار را انجام می‌دهد که economic of scale برایش وجود داشته باشد. مثلاً سایپا چون برای کل محصول مواد اولیه می‌خرد، برایش به صرفه است که برای هر محصول این کار را انجام دهد.

## Strategic Issues- Local vs. Global View

یک خریدار  $Z^*(B)$  و یک فروشنده  $Z^*(S)$  وجود دارد. اگر دنیایی وجود داشت که این دو، یکی بودند، یک entity واحد وجود داشت و من و تویی بین آن‌ها وجود نداشت. در صورتی که جدا بودند هر دو جداگانه می‌خواستند سودشان را ماکزیمم کنند و وقتی یکی باشند، شرکت واحد می‌خواهد سودش را ماکزیمم کند. می‌توان نشان داد که وقتی دو شرکت یکی باشند، سودشان بیش‌تر از جمع سود دو شرکت وقتی جدا باشند، می‌باشد.

$$Z^*(S) + Z^*(B) < Z^*(SB)$$

این نگاه local در برابر global می‌باشد که اگر global به مسئله نگاه کنیم، سودآوری می‌تواند خیلی بیش‌تر باشد. ولی در عمل شرکت‌ها خوب است که استقلالشان را حفظ کنند تا core competency اشان حفظ شود ولی جواری با هم عمل کنند که انگار، یکی هستند. یعنی بده بستان دو شرکت مستقل که دنبال سود و زیان

خودشان هستند، در عین استقلال وقتی کالا رد و بدل می کنند جوری رد و بدل کنند که سودشان به حالت global view نزدیک شود. این کار با contract ها می باشد.

ورودی ها:

C: cost of producing one unit product

P: price of selling one product

S: Salvage value of one unit product

(وقتی یک محصول، فصلش تمام می شود و بخواهیم آن را رد کنیم برود، به آن salvage value گفته می شود)

این جا مسئله ی روزنامه فروش مطرح است. چون روزنامه تا آخر روز اعتبار دارد. همیشه روزنامه فروش، دغدغه ی این را دارد که هر روز چه تعداد روزنامه برای فروش بخرد. به قیمتی که روزنامه بعد از اعتبارش خریداری می شود salvage value گفته می شود. خیلی وقت ها، salvage value حتی، هزینه ی تولید را تأمین نمی کند ولی بهتر از این است که دور ریخته شود.

اگر تعداد روزنامه خریداری شود، کم یا بیش از حد نیاز باشد، دو هزینه مطرح است:

Cost of over-stocking the product= CO= C-S

Cost of under-stocking the product= CU= P-C

می خواهیم ببینیم که Customer Service Level (CSL) بهینه چه قدر می باشد. در واقع سوال این است که من manufacture چه مقدار سفارش داشته باشم، که سود بهینه داشته باشم.

اگر کمتر از تقاضای بازار، سفارش داده باشیم، سود ما  $Q \cdot C_u$  می باشد ولی اگر بیش تر باشد سود ما می شود  $Q \cdot C_u - (Q-x) \cdot C_o$ . اگر رابطه ی ریاضی نوشته شود، حالت بهینه برای  $CLS^* = 1/(1+C_o/C_u)$  می باشد. این می تواند، یک حالت بهینه باشد.

در این مثال، اگر دو شرکت یکی نباشند، می توان یک شرکت را دید که سود آن خیلی بیش تر است. به این پدیده double marginalization می گویند. یعنی وقتی شرکت ها جداگانه سودشان را می خواهند maximum کنند، در نهایت efficient نمی باشد. یعنی هر کس local تصمیم بگیرد می خواهد خودش را local حفظ کند.

Buy-Back Contract:

در مثال قبل، از دید game theory، retailer لیدر و manufacturer فالوور می باشد. در واقع retailer تصمیم برای سفارش ۱۰۰۰ تا گرفت، که سود manufacturer تابع او بود. در این مثال buy-back زمانی رخ

در contract ها، جوری باید contract را تعریف کنیم که شرکت ها با حفظ استقلال، سودشان نزدیک به global باشد.

می‌دهد که تولیدکننده می‌گوید که تو نگران نباش، اگر نتوانستی محصول را بفروشی، من از تو مثلاً محصول را ۵ دلار می‌خرم. و سعی می‌کند که  $Q^*$  را به سمت  $Q^*$  حالت  $global\ view$  نزدیک کند.

#### :Quantity Flexibility Contracts

قبل از شروع فصل،  $manufacturer$  چون دسترسی به بازار ندارد و قدرت  $retailer$  به دلیل دسترسی به  $market$  زیاد است، می‌تواند تقاضا را تخمین بزند.  $retailer$  به  $manufacturer$  می‌گوید که من در این فصل به اندازه‌ی مثلاً ۱۰۰۰ از تو خرید خواهم داشت. تولیدکننده قول می‌دهد که اگر بیش از ۱۰۰۰ تا بیست و پنج درصد بیش‌تر خواست (۱۲۵۰)، تأمین کند و در برابر این موضوع، تأمین‌کننده هم قول می‌دهد که دیگر بیش از بیست درصد کم‌تر از ۱۰۰۰ تا، سفارشش نشود. (حداقل ۸۰۰ عدد)

#### :Revenue Sharing Contracts

در این حالت  $manufacturer$  با هزینه‌ی کم‌تری محصول را به  $retailer$  می‌دهد و درصدی از درآمد  $retailer$  را می‌گیرد. فروش امانی، نیز چنین حالتی است.

#### :Vender Management Inventory Contracts (VMI)

شرکت‌ها برای تولید نیاز به خرید مواد اولیه دارند. در چنین فضایی به جای اینکه خود شرکت مدیریت انبار کند، به  $supplier$  می‌گوید که من دیگر موجودی‌ام را مدیریت نمی‌کنم و تو بیا  $inventory$  من را خودت مدیریت کن. در اینجا مثلاً اگر یک قطعه قبلاً ده هزار تومان بود الآن پانزده هزار تومان می‌شود. این جا شرکت خریدار یک حداقل و حداکثری را به تأمین‌کننده اعلام می‌کند. دیده می‌شود که این موضوع می‌تواند عملکرد را به حالت  $global$  نزدیک کند. ایران خودرو مثلاً برای این کار نحوه تولیدش را به قطعه‌ساز  $visibility$  می‌دهد. چون قطعه‌ساز نیاز به مدیریت موجودی دارد. این فرآیند زمانی جواب می‌دهد که  $single\ supplier$  داشته باشیم. اگر چند تأمین‌کننده باشد هر کدام از  $supplier$ ها باید چندین  $min\ max$  تعیین کنند که این منجر به اضافه شدن موجودی می‌شود. متأسفانه در صنعت خودروی ما،  $multi\ supplier$  در حال انجام  $VMI$  است.

وقتی دو شرکت جداگانه عمل می‌کنند هر کدام به دنبال افزایش سودآوری خودشان هستند. به دلیل وجود داده‌ی  $asymmetric$  برای خریدار، همیشه سعی می‌کند که بیش از حد نیاز موجودی ذخیره کند ولی در حالی که مدیریت موجودی را خود تأمین‌کننده مدیریت می‌کند این اتفاق رخ نمی‌دهد و این به نفع همه است.

اگر مدیر به معنای واقعی مدیر باشد، سریع می‌فهمد که در دنیای  $exucation$  چه اتفاقی رخ می‌دهد.

$buyback$  که در این قراردادها گفته می‌شود با آنچه در صنعت نفت و گاز گفته می‌شود متفاوت است. در  $context$  نفت و گاز، یک شرکت مثلاً می‌آید یک پالایشگاه می‌زند و تا سه سال درآمد آن را برای خودش بر می‌دارد، ولی در  $context$  قرارداد بین تولیدکننده و  $retailer$ ، تولیدکننده یک قیمتی را به  $retailer$  می‌گوید که اگر محصول را نفروختی با یک قیمتی می‌خرم.

در دنیای  $exucation$  ممکن است که اتفاقات زیاد دیگری رخ دهد. ممدآفاهای خاصی ممکن است که در هر شرکت باشد که ممکن است بتواند قراردادهای میلیاردي را تغییر دهد. اینگونه ممکن است که شرکت با یک شرکت دیگر  $VMI$  بسته است ولی این فرد در شرکت از یک تأمین‌کننده‌ی دیگر خوشش بیاید و اختلال ایجاد کند.

در revenue sharing، مثلاً دیده می‌شود که ebay به افراد می‌گوید شما بیا بید بفروشید و قسمتی از درآمد را با هم تقسیم می‌کنیم.

### Purchasing Paradigms:

یک شرکتی از تعدادی از تأمین‌کننده‌ها کالا تأمین می‌کند، اگر تعداد این تأمین‌کننده‌ها بیش از یک مقداری باشد، بهتر است که این تأمین‌کننده‌ها cluster شوند و مثلاً برای کالاهای الکترونیک یک دپارتمان تشکیل شود و تعدادی تأمین‌کننده را در یک دسته قرار داد. بعضی وقت‌ها این قدر تأمین گسترده می‌شود که یک شرکت را به عنوان شرکت خریدار، شرکت تأسیس می‌کند (CB). مثلاً سایکو برای ایران خودرو. اگر در اینجا رابطه یک به یک (one to one) باشد، این corruption ایجاد می‌کند و فساد ایجاد می‌شود. همیشه بین دو شرکت، دو دوست، زن و شوهر و پدر و پسر، همیشه باید خوف و رجا وجود داشته باشد. اگر پسر بداند که اگر در مدرسه هر کاری بکند، پدرش می‌آید و معلم را می‌شورد می‌گذارد کنار، فساد ایجاد می‌کند. در این رابطه که یک شرکت تأمین‌کننده وجود دارد و باعث شده است که صنعت خودروی ایران پیشرفت نکند، این موضوع است که رابطه یک به یک است. پس بهتر است که رابطه many to many باشد تا efficiency حاصل شود.

network theory می‌تواند وجود واسطه‌ها را در کسب و کارها بررسی کند و اهمیت آن‌ها را نشان دهد.

خارج از کشور، قطعه‌سازها با چندین خودروساز کار می‌کنند و از نظر سایز و درآمد به خود خودروسازها طعنه می‌زنند. در ایران چون خودروساز power دارد، همیشه فساد ایجاد می‌کند. هر چه قدرت زیاد باشد و balance نباشد، فساد ایجاد می‌شود. خودروساز در ایران همیشه به قطعه‌ساز فشار می‌آورد و مثلاً پولش را به موقع نمی‌دهد.

آنچه در business مهم است، مدیریت روابط است. در اقتصاد که امروزه خیلی مطرح است این است که contract is incomplete، و می‌گوید هر چه قدر هم شما حواس جمع باشید، اگر طرف مقابل بخواهد بدجنسی کند و دبه کند، باز هم روشی را پیدا می‌کند. آدم‌ها بسیار پیچیده‌تر از قراردادها و سازمان‌ها می‌باشند و هر چه قدر if then condition در قراردادها بگذارید، در نهایت کم است. آنچه که مهم است مدیریت این روابط و partnership است.

انواع رابطه‌ها:

سه نوع رابطه وجود دارد:

Vertical Integration: همان merge شدن است و این که یک نفر را کاملاً takeover کنند. مانند این است که یک خودروساز برود یک قطعه ساز را بخرد.

Arm's Length Relationship: ببینیم که در مارکت، چه کسی بهتر از همه است و از او خرید کنیم. رابطه‌ی خاصی ایجاد نمی‌شود و مانند دو غریبه با هم می‌مانند و در فصل بعد دوباره در بازار می‌گردیم ببینیم چه کسی بهتر است.



Partnership of Alliance: در واقع چیزی ما بین دو حالت قبل است.

در vertical integration :

در این حالت coordination ماکزیمم می‌شود. در اینجا مشکل هماهنگی و coordination می‌تواند حذف شود و هماهنگی maximize می‌شود. چیزی که در این فضا قربانی می‌شود specialization است. در این جا ممکن است که دو ماهیت و دو طرز تفکر ایجاد شود و مثلاً یکی به R&D اعتقاد داشته باشد و دیگری نداشته باشد و در نهایت efficiency کم می‌شود.

در حالت Arm's Lenght Relationship ، هماهنگی قربانی می‌شود و مرتب باید با افراد مختلف هماهنگی انجام داد و به دلیل عدم وجود اعتماد، یک سری هزینه‌های دیگری به وجود می‌آید ولی specialization زیاد است. در واقع این جا یک transaction cost وجود دارد که باید دید در این quarter چه کسی تکنولوژی برتر را دارد و چه کسی بهتر است. در واقع از این منظر کمی این مورد دچار مشکل است. در کل در این فضا specialization ماکزیمم می‌شود اما coordination قربانی می‌شود.

رابطه‌ی سوم، سعی می‌کند از مزایای دو رابطه‌ی گفته شده بهره‌مند شود و از معایب آن‌ها پرهیز کند. در حوزه‌ی IT این مورد در حال گسترش است.

روندها بدین شرح است:

1. Advancement in IT
2. Growth in knowledge and increased complexity
3. Increased Customization

به خاطر پیچیدگی بالا و افزایش customization به شدت نیاز به هماهنگی وجود دارد و به خاطر توسعه‌های IT و افزایش پیچیدگی نیاز به specialization وجود دارد که رابطه‌های گفته شده‌ی قبلی این دو را نمی‌توانند تأمین کنند.

چنین چیزی نیاز به alliance و partnership دارد. امروزه دیگر core competency شرکت‌ها داشتن یک تکنولوژی خاص نمی‌باشد و هر چیزی که tangible باشد، قابل کپی کردن باشد، مفید نیست. این که بتوانیم با شرکت‌های مختلف رابطه داشته باشیم و با شرکت‌های متفاوت تعامل کنیم، مهم است که اصلاً کار ساده‌ای نیست و امروزه بسیار مهم است.

چه چیزی است که می‌تواند قوام یک رابطه را تأمین کند؟ عدالت. کسی می‌تواند رابطه‌ی sustainable داشته باشد که یک رابطه‌ی عادلانه را برقرار کند. وقتی رابطه‌های بازی را ساده کنیم نهایتاً به فرمایش امام علی می‌رسیم که هر چه برای خودت می‌خواهی برای دیگران هم بخواه. اگر تویوتا توانسته است که در این حوزه پیشتاز باشد نیامده است که بگوید من یک قرارداد هوشمندی نوشته‌ام که اگر دیگری جُم بخورد من بتوانم

جریمه‌اش کنم بلکه یک قراردادی شکل گرفته است که آن رابطه در آن به چشم می‌خورد. این که من چه تعداد کالا تولید کنم، زمانش در دنیای جدید گذشته است و این مهم است که بتوانیم با دیگران تعامل کنیم. یکی از مشکلات در دوران پسابرجام این است که ما یاد نگرفته‌ایم که رابطه برقرار کنیم. در هیئت مدیره‌ی بسیاری از شرکت‌ها دیده می‌شود که فقط اقوام هستند و ما هنوز نتوانسته‌ایم مدیریت و مالکیت را از همدیگر جدا کنیم. در کشور ما زیرساخت حقوقی و قانونی مشکل دارد و در این فضا، تعامل و رابطه خیلی بهتر می‌تواند مفید باشد. در رابطه‌ها ما وارد territory هایی می‌شویم که نباید وارد شیم و مثلاً همیشه قسم‌های مذهبی می‌خوریم. اخلاق، چیزی فرادینی است و باید یاد بگیریم که چگونه بدون کار داشتن به خدا و پیغمبر و مایه گذاشتن از آن‌ها (قسم خوردن های الکی)، اخلاقی عمل کرد.

اینجا مثال تقسیم یک کیک ناهمگون با طعم‌های مختلف مطرح است که اگر بخواهند تقسیم کنند ممکن است که فردی بیايد utility کیک را عوض کند و بگوید این قسمت کیک برای سلامتی مهم است و دیگری بیايد کیک را مش‌بندی کند (نگاه صناعی) و هر قسمت را بر اساس تعداد مش تقسیم کند. ولی در اقتصاد یک نگاه ساده وجود دارد و آن این است که یک نفر با چاقو کیک را ببرد و دیگری ابتدا انتخاب کند. در اینجا نه نیاز به قسم حضرت عباس وجود و نه چیز دیگر. در اینجا نفر اول می‌داند که نفر اول نیست که انتخاب می‌کند و در نتیجه سعی می‌کند جوری ببرد که utility خودش که دوم انتخاب می‌کند را ماکزیمم کند. در این حالت هر دو احساس خوبی دارند.

مثال دیگر این است که یک زمین به دو برادر به ارث رسیده باشد. یک سمت آن، ممکن است که یک بلوار باشد، سمت دیگر یک بزرگراه، سمت دیگر مدرسه‌ی شلوغ، سمت دیگر پادگان و سمت دیگر پارک باشد، این زمین را می‌توان تجاری ساخت، مسکونی ساخت و.... راه حل: نفر سومی باید به عنوان واسطه باشد. به نفر اول گفته می‌شود که آقای A هر جور که دوست داشتی زمین را تقسیم کن تا برایت fair باشد منتها برایت علی‌السلویه باشد که چه قسمت چپ برسد چه قسمت راست و این را به نفر سوم می‌گویند. نفر دوم هم، چنین کند. با این روش می‌توان زمین را به گونه‌ای تقسیم کرد که بسیار عادلانه است. (حداقل یکی از بخش‌هایی که هر کدام خواسته است، به آن‌ها می‌رسد)

آنچه در این رابطه‌ها مهم است، این است که بتوانیم عادلانه بازی کنیم.

#### Strategic Issues – SRM

در مدرسه‌های کسب و کار، معمولاً تیم‌ها تشکیل می‌شوند و یک کسب و کار را راه می‌اندازند. اما در ایران همه به صورت individual نگاه می‌کنند، و معمولاً تیم کسب و کار پایدار شکل نمی‌گیرد.

Contract کافی نیست و دو طرف رابطه‌های بلندمدت باید ایجاد کنند.

خودروسازهای آمریکایی با هم partnership قوی ندارند و اکثراً مثل غریبه‌ها عمل می‌کنند. اما خودروسازهای ژاپنی ۴۸ درصد از کل هزینه هایشان در بخش partner suppliers می‌باشد و رابطه‌شان خیلی بیش‌تر از آمریکایی‌هاست.

گفته می‌شود که هر چه قدر پیچیدگی زیادتر می‌شود، partnership بسیار اهمیت زیادی دارد. حتی دیده می‌شود که شرکت هواپیمایی بوئینگ هم، حتی علاوه بر ساخت، طراحی را برونسپاری کرد. معمولاً خودروسازهای آمریکایی در فرهنگشان خیلی سازگار نیستند.

اینکه بدانیم کدام یک از محصولات بهتر است از طریق arm length تأمین شود و این که کدام از طریق پارتنرشپ می‌تواند تأمین شود، مهم است. مثلاً توپوتا تمام اجزای خودرو که complexity دارند، از طریق partnership تهیه می‌کند ولی commodityها را از بازار به روش arms length تهیه می‌کند.

برونسپاری به صورت ماژولار است. یعنی یک سیستم را decompose کنیم اجزاء را و هر کدام را به صورت یک ماژول بدهیم به دیگران تا انجام دهند. می‌توانیم برونسپاری را بهتر انجام دهیم. مثلاً مارکتینگ خیلی پیچیده است و اگر بخواهیم یک قسمتش را برونسپاری کنیم، شاید کار پیچیده‌تر شود.

اگر کسی که پارتنرشپ با او نداریم و یک فرآیند را به او بدهیم، می‌تواند این کار را یاد بگیرد و جایگزین ما شود. هر چه قدر محصول پیچیده‌تر باشد، interdependency زیاد می‌شود و مجبوریم که در دپارتمان‌ها joint انجام بدیم.

شرکت کرایسلر:

ترمینولوژی شرکت برای همکاری، یا transaction است، یا coordination یا cooperation یا alliance. و سطح این نوع همکاری در سه سطح Basic، duration و integration می‌باشد.

در حالت transaction، پایه‌ی همکاری در حد ارائه‌ی اطلاعات است و هیچ integration ای هم لازم نیست. Data فقط مبادله می‌کنیم.

در رابطه coordination، در حالت بیسیک information تبادل می‌شود و در سطح بعدی قراردادها شکل می‌گیرد و گفته می‌شود که مثلاً ما می‌خواهیم شما قسمتی را برای ما طراحی کنید. در حالت بعدی بررسی می‌کنند که چگونه با همدیگر مثلاً طراحی را انجام دهند. در این سطح، پیش‌بینی می‌شود فصل بعد میزان فروش فلان ماشین چه قدر است و به تأمین کننده گفته می‌شد.

در رابطه cooperation یک know-how بر قرار می‌شود. مثلاً نحوه‌ی طراحی گیربکس را هم به تأمین‌کننده‌اش می‌گوید. Life cycle کار می‌کنند، برای این نسل از گیربکس، طراحی کنید. یکپارچگی بالا می‌رود.

زمانی می‌توانیم کار را به دیگری محول کنیم که دیگری با ما خودی شده باشد و سر سفره‌ی ما باشد.

در رابطه alliance، یکی نمی‌شوند ولی بسیار رابطه قوی می‌شود و با هم قاطی می‌شوند. حتی در بخش‌های فروش هم با همدیگر رابطه برقرار می‌کنند. باید یاد گرفت که بتوانیم فضای رابطه ی alliance را مدیریت کرد.

در طراحی، هر چیزی را برونسپاری نمی‌کنند، بلکه detailed design را برونسپاری می‌کنند. در کانتکست IT وظیفه ی system integrator این است که بخش‌های مختلف را با هم هماهنگ کند. شرکت‌هایی که طراحی را برونسپاری می‌کنند، بخش system integrator را در دست خودشان گرفته‌اند.

با برونسپاری detailed designها، دیگر طراحی اجزای مختلف ماشین‌ها مثل هم می‌شوند، برای همین است که دیگر core competency در این موارد tangible نیست. در مدیریت زنجیره تأمین، change management مطرح است. این‌که سریع‌تر نیاز بازار را بشناسد، time to market اش کم‌تر باشد، برنده است.

#### :Traditional relationship

آنچه به صورت سنتی در رابطه‌ها مهم است، یکی قیمت است که بسیار مهم است. عرضه کننده متولی اصلی کیفیت است و عملاً خریدار کنترل می‌کند که کیفیت مناسب می‌باشد یا خیر. یک مدل ذهنی چانه‌زنی وجود دارد که عرضه‌کنندگان را به جان هم بیاندازیم. چون همیشه عدم اعتماد وجود دارد، خریدار می‌خواهد که خودش را از این موضوع حفظ کند و این هزینه را بالا می‌برد. اما در فضاهای جدید، می‌تواند یک فضا سافت‌تر باشد که فقط قیمت مهم نیست و موارد زیاد دیگری هست. نظیر این که کیفیت، تحویل به موقع و .. و قیمت مطرح است، بر خلاف نگاه سنتی که فقط قیمت مهم است، موارد زیاد دیگری مهم است. باید دید که تأمین کننده چه قدر قابلیت‌های مدیریتی، قابلیت‌های مدیریت انسانی، مدیریت تولید دارد و مدیریت روابطش چگونه است؟ باید دید با شرکتی می‌خواهیم کار کنیم که مدیر معتقد است که می‌خواهد دو روزه پولدار شود و برود یا این که می‌خواهد مدت‌ها در این صنعت بماند؟ آیا به افراد دید برده دارند؟ آیا من با کسی کار می‌کنم که نیروی انسانی‌اش حاضر است چون بدهد برای کسب و کار یا این که از مدیران نفرت دارد؟

#### از منظر management capabilities:

آیا مدیران به اندازه‌ی کافی تجربه دارند یا این که پیشرفت را به صورت آسانسوری طی کرده‌اند؟ یا این که همین طور دارند کار می‌کنند و اصل پولشان را دارند از real state در می‌آورند. این خیلی رایج است (شرکت‌های نرم‌افزاری مثلاً) که درآمد اصلی‌شان از چیزی غیر از عملیات اصلی‌شان است. آیا جناب کاله از محصولاتش پول در می‌آورد یا چون احتیاجات اولیه مردم را تأمین می‌کند در آمل زمین بزرگی به او می‌دهند و بارش را آن جا می‌بندد؟ اگر مدیری بلندمدت نبیند اصلاً سرمایه گذاری نمی‌کند. برای مشکلات آینده چه قدر می‌خواهند آماده شوند؟ چه قدر می‌توان روی حرف آن شرکت حساب کرد. قدیمی‌ها وقتی یک فرش را می‌فروختند و روی یک قیمتی توافق می‌کردند دیگر این قیمت ثابت می‌ماند.

بوئینگ طراحی general را نگه می‌دارد و detailed design را برونسپاری می‌کند. بوئینگ به عنوان system integrator عمل می‌کند.

از منظر personal capabilities:

چه قدر تحصیلات دارند؟ اخلاق چگونه است؟ چه قدر سازمان را ترک می‌کنند؟ آن‌ها چه قدر به مفاهیمی مثل کیفیت پایدار هستند؟ رابطه اشان با کارفرما چگونه است؟ مثلاً چینی‌ها با این که با حجم بالا تولید می‌کنند ولی اکثراً از نیروی انسانی مانند برده‌ها استفاده می‌کنند. در این فضا می‌توانند یک دستگاه ریش‌تراشی تولید کنند و در والمارت آمریکا بفروشند ۴,۹۹ دلار. آیا این کار درست است که با آدم‌ها این گونه برخورد کرد؟

بازدید از کارخانجات، جنبه‌های مختلف دارد. مثلاً تویوتا پیوسته تور می‌گذاشت که حتی رقبا بیایند کارخانه را ببینند و دیگران می‌گفتند که بقیه می‌آیند کار را یاد می‌گیرند. ولی مزیتی که داشت این بود که همیشه محیط کارخانه تمیز نگه داشته می‌شد. همین‌طور سعی می‌کردند که فضای کارخانه را به سمتی ببرند که فضا فضای دوستانه شود.

از جنبه‌های تولید:

در سیستم تولید چه قدر استاندارد وجود دارد؟ سیستم planning چگونه است؟ نگهداری آن‌ها چگونه است؟ ما ممکن است که خیلی از این‌ها را داشته باشیم. مثلاً ایران‌خودرو انواع استانداردها را دارد ولی ماشین که تولید می‌شود دنده‌اش جا نمی‌خورد. باید دید که سابلایر آیا certifate دارد یا نه؟ البته باید مواظب بود که واقعاً دارد یا نه و به درد می‌خورد یا خیر.

قابلیت‌های رابطه‌ای:

به جای این که ما محصول را بعد از تولید، کنترل کیفیت کنیم، در مکان حضور داشته باشیم و رابطه در این حد قوی باشد که بتوانیم در مکان‌های همدیگر حضور داشته باشیم.

Kpi هایی که برای supplier performance measuerement مطرح است، همان چیزهایی است که قبلاً گفته شده بود. کیفیت، هزینه و.....

supplier development: به این معنا است که تمام فعالیت‌های که خریدار انجام می‌دهد تا قابلیت‌های سابلایرش را تقویت کند. مزایای این کار این است که هر چه قدر ما بتوانیم سابلایر را تقویت کنیم، سودش به خودمان می‌رسد. مثلاً اگر IT سابلایر را تقویت کنیم، خودمان بهره می‌بریم. عیب این است که ما سرمایه‌گذاری کنیم که با ما بهره ورتر شود ولی بعد سوئیچ کند به رقبا. یک نگرانی مهم در این موضوع این است که اگر ما سابلایر را ارتقا دهیم ممکن است که او برود و با رقیب کار کند. اگر از نگاه خشک اقتصادی نگاه کنیم این موضوع می‌تواند مهم باشد ولی از نگاه سعه‌ی صدر، این موضوع مهم‌تر است. در بلندمدت این به نفع ماست که این کار را انجام دهیم. supplier development بر اساس بده بستان دو دوتا چارتا نیست این که من این کارها را برای انجام می‌دهم تا تو این کار را انجام دهی برای ما. معمولاً در نگاه سعه صدر این طور عملاً

می‌کنند که مثلاً من سیستم آی تی تو را ارتقا می‌دهم ولی با سیستم آی تی خودم اینتگریتد می‌کنم. در واقع سعی می‌کنیم که در *supplier development* بیش‌ترین سود را برای خودمان بخواهیم. مثلاً ژاپنی‌ها در آمریکا بعضی از خودروهایشان را تولید می‌کنند و سیستم تولید در آمریکا را با سیستم تولید خودشان سنکرون می‌کنند و مثلاً بچ‌های ۱۵ تایی در دستور کار دارند در صورتی که دیگر خودروسازها بچ‌های ۲۰ تایی دارند و یک استاندارد خاص ایجاد می‌کنند.

در نگاه گیم تئوری، اگر دید *repeated game* وجود نداشته باشد، معمولاً *cooperation* رخ نمی‌دهد.

*supplier development - process map*: یک سری برنامه‌ریزی شکل می‌گیرد.

*supplier development – pitfalls*: اگر از سمت ساپلایر تعهد کافی نباشد و همین طور از دید خریدار.

بررسی کیس دریملاینر:

برخلاف ایرباس، اقتصادی‌بودن بوئینگ مطرح بود و برای همین از مواد کامپوزیت استفاده می‌کردند و برای همین تمامی طراحی را باید از نو انجام می‌دادند. چون مصرف سوخت هواپیما کم می‌شد، در سال‌های ۲۰۰۳ یک سفارش بسیار زیادی گرفت. قرار بود که ۲۰۰۷ تحویل دهد، ولی تا سال ۲۰۱۱ اولین پرواز آزمایشی شکل نگرفت.

مدل *revenue sharing*، خیلی مطرح بود. این مسئله‌ای که در بوئینگ مطرح بوده است، مسئله‌ای است که هر روز در کشور ما تکرار می‌شود. چهارتا فارغ التحصیل می‌خواهند با هم شرکت بزنند، یا چند فامیل می‌خواهند با گذاشتن یک پولی، یک ساختمانی را بسازند. بعد از شروع این، می‌بینند که همه چیز متفاوت است و پول کم می‌آید و کارها دیرتر انجام می‌شود. در این کیس، بوئینگ یک حرکت جسورانه‌ای انجام داد که از همان ابتدا، حتی طراحی را به بیرون سپرد. قبلاً برونسپاری طراحی *taboo* بوده است چون *core competency* بوده است. *detailed design* را برونسپاری کرده است ولی *overall design* را در دست خودش گرفت.

همین طور برای مواردی مثل همراه اول وجود دارد، که مثلاً *detailed design* مثل زیرساخت‌ها را به دیگران بسپارد.

آنچه مهم است این است که *strategic positioning* حفظ شود و این سوال همیشه مطرح است که آیا این بازی را در *long term* نباخته‌ایم؟ اگر یکی از دنیای *excution* دور باشد، دوز و کلک‌ها را فراموش می‌کند و مشکل *obsolete* شدن دانش در برونسپاری بسیار مهم است و باید توجه کرد که از دانش عقب نیفتیم. کسی می‌تواند به هندی‌ها در نرم افزار برونسپاری کند که خودش *programmer* حرفه‌ای باشد. ولی اگر خودتان هیچ چیزی مثل *java* و... ندانید، برونسپاری معنا ندارد. کسی می‌تواند برونسپاری کند که خودش بتواند آن برونسپاری را مدیریت کند. این که چگونه مدیریت کنیم و تا کجا و تا چه عمقی لازم است، می‌توان به تیم برونسپاری رفت، *job rotaion* انجام داد و تیم‌های مهندسی خودمان و دیگران را کنار هم بنشانیم. وگرنه اگر بخواهیم همه چیز را از اول تا آخر برونسپاری کنیم، دانشمان *obsolete* می‌شود. بعضاً برای مدیران ارشد کسانی از خارج از سازمان باید بیایند تا تکنولوژی روز را به مدیران ارشد انتقال دهند. کلیات و روندها را باید دانست و بعد برونسپاری کرد. یک معضل برونسپاری این است که دانش *obsolete* می‌شود و باید جلوی آن را گرفت.

مشکل *obsolete* شدن دانش در برونسپاری بسیار مهم است و باید توجه کرد که از دانش عقب نیفتیم.

در برونسپاری کیس بوئینگ: وقتی مثلاً موتور را به کسی برونسپاری کنیم، ممکن است که هوس کند که روزی هواپیما بزند. یا کسی که بال هواپیما می زند، ممکن است که هوس کند که هواپیما بسازد. وقتی detailed design را به دیگران می دهیم، به نوعی نقش system integrator داریم. کسی نمی داند که چگونه سیستم هیدرولیک با سیستم برق هماهنگ می شود. در نهایت ما overall را داریم. هر چه قدر که در detail خبره شود، نمی تواند به overall دسترسی داشته باشد. مثل این است که یک شرکت به شدت در ترمز abs حرفه ای شده باشد، نمی تواند خودرو بسازد.

باید چیزی را که برونسپاری می کنیم، این قدر آن چیز را بلد باشیم که مدیریت کنیم.

در این کیس، به دلایل مختلف ذکر شده در کیس، delay شکل می گیرد.

در این کیس، یک هواپیمای که طراحی بسیار پیچیده بود، احتمال به مشکل خوردن زیاد بود. انواع delay رخ داد. مسائل فرهنگی در نظر گرفته نشد. مثلاً ژاپنی ها جور دیگری کار می کنند. مثلاً حتی اگر کارش به خوبی پیش رفته باشد، می گوید کلی کار دیگر داریم ولی ایرانی ها با ده درصد پیشرفت کار را کامل می دانند.

در مثال ساختمان سازی بالا، وقتی به مشکل خورد، و نیاز به پول های بیش تری باشد، هر کس به دیگری نگاه می کند که اگر دیگری پول داد من هم پول می دهم و در نهایت، هیچ کس پول نمی دهد و منتظر است تا دیگری پولی بدهد.

در کیس بوئینگ قرارداد که revenue sharing است، وقتی به مشکل خورد، بازی free riding رخ می دهد و هر کس می گوید که دیگری باید کاری کند و برای سرمایه گذاری مجدد، همه می خواستند کاری کنند که تقصیر را بر دیگری بیندازند. پروژه که به تأخیر می خورد و بقیه می خواهند چک بدهند، اگر یک نفر مثلاً مهر بتواند پول بدهد، بقیه هم برای مهرماه چک می دهند. این تأخیر باعث می شود که تا خود مهر، دوباره نیاز به پول بیشتری باشد. باید از ابتدای بازی، قواعد بازی را صفر کرد. مثلاً اگر به تأخیر خورد، باید گفت که ممکن است که شریک دیگری اضافی شود. در این کیس، وقتی کسی که بال می سازد، می دید که کسی موتور را می سازد، تأخیر دارد پس او هم در سرمایه گذاری اش تأخیر ایجاد می کند و هر کس خودش را پشت سر فرد دیگری قایم می کند.

اگر بپرسیم که بوئینگ چی کار می کرد که بهتر بود؟ جواب مشخصی نیست

این که از صد درصد طراحی داخلی به صد درصد برون سپاری طراحی حرکت کرد، کار جسورانه ای بود. شاید به خاطر این که همه ی قطعات مواد کاپوزیت بوده است، باید همه چیز با همدیگر مونتاژ می شد. به نظر می رسد که اگر این radical change را خرد می کرد، شاید می توانست که بهتر عمل کند و در فازهای بعدی جزئیات را به دیگران بدهد که طراحی کند.

یک پروژه برونسپاری همراه اول:

برونسپاری و شراکت تجاری در همراه اول

در جهان جدید شاهد این هستیم که:

نیروگاه های اتمی توسط بخش خصوصی سپرده می شود.

زندان بانی برونسپاری را می کند.

امنیت و حفاظت از افراد در جنگ (عراق) توسط بخش خصوصی است.

جهان جدید باید یاد بگیرد که تعامل داشته باشد.

حوزه‌های گفته شده، نسبت به کسب و کار خیلی پیچیده‌تر است. هنر این است که چگونه قواعد بازی را مدیریت کنیم و کار را به دیگران بدهیم و خودمان روی نقطه قوت توجه کنیم.

در مقام منبع‌یابی، باید بدانیم که چرا می‌خواهیم برونسپاری کنیم؟ چه چیزی می‌خواهیم برونسپاری کنیم؟ چه کسی می‌خواهد برونسپاری کند؟ به چه کسی می‌خواهیم برونسپاری کنیم؟ چگونه می‌خواهیم برونسپاری کنیم؟ چه موقع می‌خواهیم برونسپاری کنیم؟ اصلی‌ترین بازیگر، برای برونسپاری در صنعت مخابرات، کاهش هزینه‌ها است.

وقتی می‌خواهیم یک چیز را برونسپاری کنیم، می‌تواند سه نوع باشد: ۱- کل فرآیند برونسپاری شود ۲- بخشی از فرآیند برونسپاری شود ۳- کل فرآیند نگه داشته شود و دیگران را از بیرون آورد (استخدام کرد).

## IT Value Chain

آقای بیل گیتس در سال ۱۹۷۵، در زمانی که اوج بخش‌های سخت‌افزاری بود، گفت که آینده صنعت IT در گرو نرم افزار خواهد بود.

از دو بعد می‌شود به زنجیره ارزش IT نگاه کرد. از بعد SERVICE OBJECT، سه استیج Hardware ->

Development -> Consultant Software وجود دارد. بعد دیگر IT service VC با ابعاد ->

Operation -> Implementation می‌باشد.

system integrator ها در این زنجیره ارزش، کسانی هستند که نه نرم افزار دارند و نه سخت افزار. از این قسمت می‌توان در دیگر کسب‌وکارها، الهام گرفت. در واقع بخشی از زنجیره تأمین، مشاوران صنعت یا کسب‌وکار می‌باشند. بخشی به عنوان ساپلایر و بخشی به عنوان provider می‌باشند. IBM فرآیند HR شرکت‌ها را مدیریت می‌کند و شرکت‌ها فرآیند HR را به IBM می‌سپارند. اگر خودمان صاحب تکنولوژی نیستیم، system integrator می‌تواند یک حوزه‌ی مفید باشد. نه تنها در حوزه ی IT بلکه، در مثلاً حوزه‌ی فلز، می‌تواند خیلی مفید باشد و حتی در محیط‌های صنعتی هم خیلی شایع است. کسی می‌تواند این نقش را ایفا کند که همه‌ی software ای‌ها و hardware ای‌ها را بشناسد.

یک زمانی در ایران می‌خواستند که سیستم عامل و ERP، ایجاد کنند. ولی واقعیت این است که ما این کاره نیستیم و این کار نیاز به یک R&D میلیاردی دارد.

SAP از اکو سیستم IT در حوزه‌ی App Service provider و APP SW supplier وجود دارد. می‌تواند که برود شرکت‌های مشاوره‌ای را بخرد ولی نمی‌رود. IBM به جز بخش نرم‌افزار، تقریباً در تمام حوزه‌ها حضور دارد. logo یک شرکت ترکی است که وقتی خارجی‌ها حضور نداشتند، نقش پررنگی داشت ولی اکنون ندارد. meteksan sistem یک شرکت ترکی است که در بخش integrator به خوبی عمل می‌کند.

این که چه چیزی core competency می‌باشد، به شدت تابع مدل کسب و کار است. ایرانسل دنبال low price است ولی همراه اول دنبال تنوع و کیفیت.



به شرکت‌های داخلی توصیه می‌شود وارد نرم‌افزارهای Planning صنعتی نشوند و این یک باتلاق است. Develop کردن این موضوع کار هر کسی نیست. فی‌الظاهر ساده است، اگر درهای کشور باز شود هم به کل از بین می‌رود، درها بسته باشد و بماند شاید بمانند ولی همواره در تهدید هستند.

یک System Integrator باید به طور کلی، خنثی باشد و طرف هیچ شرکتی را نگیرد تا بتواند نقشش را به خوبی ایفا کند. مثلاً شرکت کاله، یک شرکت تحقیقات بازاریابی راه‌اندازی کرد. رقبا می‌خواستند از این شرکت اطلاعات بگیرند و این شرکت نمی‌توانست که بی‌طرف باشد.

وقتی همه‌ی شرکت‌ها به سمت specialization می‌رفتند، IBM به سمت total solution حرکت کرد. system integrator در IBM مثلاً نمی‌آیند که بگویند از سخت افزار HP استفاده کن، ولی در نهایت یک راهکاری ارائه می‌دهند و آن را در حوزه‌های مختلف گارانتی می‌کنند.

در حوزه‌ی IT اگر یک نفر سخت افزار بدهد و یکی نرم افزار و این سیستم کار نکند، نرم‌افزاری می‌گوید که تقصیر سخت افزار است و برعکس و هر کدام دیگری را متهم می‌کند.

## Inventory Management

مبحث مهمی است. جریان مواد، سه حوزه را در بر می‌گیرد:

مدیریت موجودی تقریباً در تمام اجزای زنجیره تأمین مطرح است و باید مدیریت شود.

اگر مراحل مختلف زنجیره تأمین را در نظر بگیریم، از تأمین‌کننده تا مشتری، ممکن است که موجودی را با نام‌های مختلف نام‌گذاری کنیم. RM یا مواد خام، WIP یا Working in Process که چیزی است که داخل یک operation است. موجودی که در حین حمل باشد را in transit می‌گوییم.

آیا داشتن موجودی خوب است یا بد است؟ excess inventory یک اپدیمی در سازمان‌ها است و مشکل بزرگی است. وقتی یک مشکلی در سازمان‌ها داشته باشیم، این یک مشکل همه‌گیر خواهد بود. اگر مدیریت موجودی و مدیریت عملیات بلد نباشیم، می‌گوییم که از این تعداد تیر آهن، صد باکس می‌خرم تا تمام مشکلات تأمین را از بین ببرم و مدیر ارشد بسیار راضی است. این بحث در بخش تولید هم وجود دارد، که از همه چیز زیاد می‌خرد تا هیچ وقت خط تولید نخواهد. در هر جای زنجیره تأمین excess inventory می‌تواند مشکل کمبود را حل کند. این مانند مسکن می‌ماند. پزشک باید ریشه‌ی یک بیماری را پیدا کند و از بین ببرد نه این‌که برای هر درد یک مسکن بدهد. من مدیریت procurement بلد نیستم، پس از excess inventory استفاده می‌کنم. نتیجه‌ی چنین کاری در balance sheet دیده می‌شود و این‌که یک مقدار زیادی پول، خوابیده است.

موجودی نقش خون را در یک بدن دارد و همه‌ی اکسیژن دادن‌ها از طریق موجودی است.

در فضایی قرار داریم، که محصولات customized شده‌اند و تعاملات زیاد شده‌اند. بر خلاف روزگار قدیم که با داشتن موجودی بالا، می‌شد عملیات را بهتر تأمین کنیم، اکنون این طور نیست که مثلاً با گذشت زمان، یک چیز سریع obsolete می‌شود.

برای کنترل موجودی باید چهار سوال را پاسخ‌گو باشیم:

کجا؟ آیا در بخش supplier موجودی باید داشته باشیم؟ در بخش تولید؟ یا در بخش توزیع؟

چه چیز (محصول)؟

چه قدر؟ به میزانی که service level بالا رود، میزان موجودی متفاوت است. اگر بخواهیم سطح خدمت صد در صد داشته باشیم، نمودار موجودی به سمت بی‌نهایت می‌رود. افزایش موجودی، باعث می‌شود که هزینه بالا رود. در واقع با بالا رفتن سطح خدمت، هزینه بالا می‌رود و باید بین این دو یک نقطه بهینه را پیدا کرد.

Policies? سیاست‌ها چیست؟ اگر من در یک جا مشخص کرده‌ام که سطح موجودی باید صد باشد، اگر کم شد و به نود رسید، چگونه این را دوباره پُر کنم و دوباره به صد برسانم؟

AMR (American Market Research) خیلی قوی عمل می‌کند و یک گزارش سه صفحه‌ای آن هزاران دلار می‌ارزد. این شرکت گزارش داد که اگر بتوانیم به طور بهینه و global view موجودی را مدیریت کنیم، می‌توانیم هم موجودی را کاهش دهیم و هم سطح خدمت را زیاد کنیم. این موضوع در سایه‌ی global view می‌باشد.

#### لایه‌های مدیریت موجودی

در حوزه‌ی مدیریت موجودی، سه لایه مطرح است. یک سری جنبه‌ها استراتژیک است. یک سری جنبه‌های tactical و یک جنبه‌ی execution وجود دارد. service level باید چه قدر باشد، تا در موجودی سرمایه‌گذاری کرد. مسائل tactical این را نشان می‌دهد که چیدمان چگونه باشد. execution در واقع day to day operation برای مدیریت موجودی می‌باشد.

هر چه موجودی بیش‌تر در سیستم داشته باشیم، responsiveness بیش‌تری خواهیم داشت. هر چه به سمت raw material برویم، عملاً efficiency زیاد می‌شود و موجودی کم‌تر می‌شود تا هزینه‌ها کم‌تر شود.

#### انواع موجودی:

cycle inventory: خیلی از کارها به صورت batch و lot size می‌باشد. اگر یک فروشگاه مثلاً تلویزیون سونی می‌فروشد، این طور نیست که تا فروخت سفارش بدهد تا دیگری بیاید. بلکه به صورت مثلاً ده تایی با وانت می‌خرد. همین طور کسی که این مغازه از او می‌خرد، ممکن است به صورت کانتینر از خود شرکت سونی بخرد. همیشه موجودی به صورت بسته بسته مثلاً ده تا ده تا خریده می‌شود. این موجودی اجتناب ناپذیر است.

inventory را باید as much as needed داشته باشیم، نه as much as possible.

موجودی ماهیت optimization دارد. در واقع باید حالت بهینه را با توجه به هزینه و سطح خدمت به دست آورد.

در نگاه گلوبال، نه تنها می‌توان میزان موجودی را کم کرد بلکه سطح خدمت را زیاد کرد. این نگرش از سال ۲۰۰۵ شکل گرفته است.

saftay inventory: این موجودی است که به طور default وقتی می‌گویند مدیریت موجودی، این موجودی را می‌گویند. این نوع موجودی باید demand satisfaction را زیاد کند. عملاً به خاطر پاسخ‌گو بودن به تقاضای uncertain درصدی را نگاه می‌داریم.

Seasonal inventory: گاهی اوقات، تقاضا seasonality دارد. قیمت هم ممکن است که seasonality داشته باشد. کالا در یک فصل خریده می‌شود و در فصل‌های دیگر مصرف می‌شود. مثلاً فرض کنید شرکتی که اسپری حشره‌کش می‌زند، حشره‌ها بیش‌تر در تابستان هستند. در نتیجه ممکن است به خاطر تقاضا قیمت مواد خام برای اسپری در تابستان زیاد باشد و شرکت در فصل زمستان بخرد و در تابستان مصرف کند. دائم یک فضای stochastic وجود دارد و ممکن است که الگوهای خرید تغییر کند و در فصلی که قیمت پایین بوده است، تقاضا زیاد شود و قیمت بالا رود. همیشه یک راهکار hybrid باید در نظر گرفت که یک محصول خیلی زود خریده نشود و یا خیلی دیر خریده نشود.

### The role of cycle inventory in Supply Chain

موجودی متوسط است که به خاطر lot size وجود دارد. در مهندسی صنایع بررسی می‌شود که بهینه‌ی سفارش چیست.

Q: Quantity in a lot or batch size,

D: Demand/unit time

cycle inventory =  $Q/2$

flow time = Ave.inv/Ave. flow rate =  $Q/2D$

در یک سیکل سفارش (که سفارش Q داده می‌شود و در یک زمان تمام می‌شود  $Q/2$  می‌باشد)

در تئوری صف به این قضیه little's law گفته می‌شود. یعنی اگر یک سیستم صف وجود داشته باشد، باید نرخ ورود و نرخ خروج برابر باشد. برای stable ماندن سیستم، باید نرخ میانگین ورود و خروجی برابر باشد. Arrival rate = Departure rate.

میانگین زمانی که آدم‌ها در سیستم می‌مانند \* زمان تعداد آدم‌ها = میزان میانگین موجودی

برای این که سرور Down نشود، نرخ ورودی باید با نرخ خروجی برابر باشد.

اگر به طور متوسط، در هر ساعت ده نفر وارد شوند و ده نفر خارج شوند و هر نفر ده دقیقه کارش طول بکشد، در هر لحظه تعداد ۱۰۰/۶۰ نفر وجود دارد.

سیستم اگر بخواهد، LIFO باشد، سیاست کاری‌اش فرق دارد. تا وقتی که Priority Base باشد، به این‌ها می‌گویند سیاست Processing.

هر چه قدر variety بالا رود، average waiting time بالا می‌رود. پس با کاهش variety می‌توان، average waiting time را کم کرد. مثلاً در سیستم بانک، کسانی که کارشان کم‌تر طول می‌کشد، به طور جداگانه خدمات بگیرند.

انواع رویکردها برای این که ببینیم چگونه باید سفارش داد، وجود دارد.

هر چه Batch Size را کم کنیم، میانگین موجودی‌مان طبق Little's Law کم می‌شود، ولی از آن طرف هزینه‌هایی نظیر حمل و نقل، افزایش می‌یابد.

در یک مدل ریاضی، داریم:

Average price per unit purchased (C)

Fixed ordering cost incurred per lot (S)

$$TC = C * Q + \frac{D}{Q} * s + \frac{Q}{2} * h * c \Rightarrow Q^* = EOQ = \sqrt{\frac{2 * D * S}{h * C}}$$

این نگاه، نگاه لوکالی می‌باشد. پایین دست و بالادست ممکن است که این موضوع را فقط از دیدگاه خودش ببینند.

مدل‌های دیگر نیز وجود دارد که بر اساس Discount می‌باشد. هر چه قدر بیشتر سفارش دهید، تخفیف می‌گیرید.

بعضی از مدل‌ها Volume-based می‌باشند. این مدل می‌گوید که من به هر سفارش شما کاری ندارم، بلکه به مجموع سفارش شما در سال کار دارم که اگر از یک مقداری بیشتر باشد، تخفیف می‌دهم.

دو نوع تخفیف وجود دارد که یکی به کل محصولات تخفیف می‌دهد و دیگری به صورت marginal می‌باشد.



برای پیدا کردن متوسط موجودی باید دید که در سیستم چه نرخي در زمان وارد می‌شوند، چه قدر زمان انتظار است و این‌ها را در هم ضرب کنیم، میزان موجودی به دست می‌آید. (قانون little)

در حالت Marginal-Unit Quantity Discount، وقتی حجم سفارش مثلاً از پنج هزار بالا رفت، برای این پنج هزار اول، تخفیفی در نظر گرفته نمی‌شود ولی برای بیش از این مقدار، فقط به خود چیزی که بیش‌تر است، تخفیف ارائه داد می‌شود.

### Role of Safety Inventory

برای پاسخ‌گویی به تقاضای uncertain است. این نوع موجودی tradeoff دارد. در واقع مانند مسئله‌ی روزنامه فروش، cost of over-stock و cost of under-stock در این جا وجود دارد. در این مدیریت موجودی نگاه لوکال و گلوبال وجود دارد. گفته می‌شود که من در این قسمت، الگوی پایین دست و بالادست را می‌دانم.

local view پنج پارامتر بیش‌تر ندارد و به راحتی در اکس قابل به دست آوردن است. اول، چه میزان می‌خواهیم سطح خدمت داشته باشیم. با این پنج پارامتر، هر منحنی تقاضا را که در نظر بگیریم، می‌توان میزان بهینه موجودی را به دست آورد.

Fluctuation های پایین دست، می‌تواند این باشد که order تغییر کند. به طور کلی حدود ۳۰ درصد تغییر می‌کنند. از سمت بالادست مواردی نظیر زیاد شدن leadtime مثلاً به خاطر برف، یا دیر شدن delivery یا مثلاً خراب شدن دستگاه تأمین کننده می‌تواند باشد.

### Measuring product availability (CSL)

دو حالت دارد:

product fill rate: میزان کمبودی که ممکن است در تعداد کالا رخ دهد.

order fill rate: من به quantity کاری ندارم، به تعداد order که می‌توانم پاسخ گو باشم، کار دارم.

stock out: احتمال این که من در سیستم موجودی نداشته باشم، چه قدر است؟

فرض کنید که ۸۰۰ موجودی داریم، سفارش اول، ۹۰۰ عدد و سفارش دوم ۱۰۰ باشد. طبق تعریف اول CSL=80% می‌باشد ولی در تعریف دوم، CSL=50% می‌باشد. با این که، تعریف اول، منطقی‌تر به نظر می‌رسد، ولی در صنعت تعریف دوم، بیش‌تر استفاده می‌شود.

در کل اگر در فضای commodity business هستیم و به بازار می‌فروشیم و اصلاً مشتری برایم مهم نیست، تعریف اول در این گونه محصولات و صنایع توجیه دارد. اما وقتی مشتری یا محصول خاص می‌شود، تعریف دوم مهم است. مثلاً در صنعت غذایی اگر یک مشتری با تعداد پنج نفر بیاید و فقط بتوان به سه نفر غذا داد، باعث نارضایتی آن‌ها می‌شود.

روندها به این سمت است که تعریف دوم، بیش‌تر اهمیت پیدا می‌کند.

local view (Buffer centric)  
آنالیزهای آماری

است.

Global view (network centric)  
از جنس بهینه‌سازی

است.

$CSL^* = 1/(1+Co/Cu)$  است. ولی حساب کردن  $Cu$  سخت است. در سیستم‌های IT، خیلی وقت‌ها  $Cost of Understock$  معلوم نیست. چون سفارش شکل نمی‌گیرد. خیلی وقت‌ها از جنس  $Cost process$  می‌باشد. گرچه تئوریکالی کاملاً معتبر است.  $Co$  برای حساب کردن راحت‌تر است. چون می‌دانیم چه قدر موجودی اضافه آورده‌ایم.

$Co/Cu$ . اگر این دو با هم برابر باشند، و هر منحنی تقاضا داشته باشیم، اگر متوسط را در نظر بگیرم،  $CSL^*$  به دست می‌آید. اگر تقاضا رندم می‌باشد، باید دید که نسبت  $Co$  به  $Cu$  چه قدر است؟

در صنعت معمولاً شرکت‌ها  $ABC$  classification انجام می‌دهند.

### Replenishment Policies

برای replenishment دو پارامتر نیاز داریم. یکی این که با چه تعدادی و در چه زمانی سفارش دهیم. در دنیای واقعی دو سناریو وجود دارد که همیشه یکی از این‌ها کار می‌کند. گاهی اوقات  $Q$  را ثابت می‌کنند و زمان متغیر است (Continuous Reveiw) گاهی اوقات زمان ثابت است و  $Q$  متغیر است (Periodic review).

در حالت Continuous همیشه یک هزینه برای مانیتور کردن میزان موجودی وجود دارد. در حالت periodic review چون اطلاعی از میزان موجودی نداریم، همیشه سعی می‌کنیم که موجودی بیش‌تری داشته باشیم. (ترس از صفر شدن موجودی)

خیلی وقت‌ها periodic درست است که موجودی بیش‌تری وجود دارد، ولی practicalتر است. چون که آمادگی ذهنی وجود دارد و صنعت به لحاظ روانی با periodic راحت‌تر عمل می‌کند. برای ساده کردن حتی  $Q$  را ثابت می‌کند و می‌گوید که هر شنبه، صد تا سفارش بده. این از حالت بهینه کاملاً دور می‌شود ولی خیلی practicalتر است.

سازه گستر، تأمین سایپا را انجام می‌دهد. سازمانی که متولی تأمین است، یک سری فرآیندهای مدیریتی دارد که پیش‌بینی بازار را بر می‌گیرد. فرآیندهای مرکزی مانند تحویل کالا به مشتری است و فرآیندهای پشتیبانی مانند خدمات پس از فروش است.

### ABC classification

نوعی از این کار نسبت  $cost of understock$  به  $cost of overstock$  را بررسی می‌کند. نداشتن محصول ممکن است که نسبت به  $opportunity cost$  موجود به برند یک ضرر زیادی بزند.

در سگمنت نیمه رساناها، بررسی کردند پنج درصد محصولات، سی درصد سفارش‌ها را شامل می‌شده است و پنجاه درصد درآمد را از آن خود می‌کرده است. شصت درصد حجم کار برای این محصولات بوده است.

در مقام مدیریت موجودی اگر همین پنج درصد را به خوبی مدیریت کنیم، تقریباً شصت درصد حجم کارها را مدیریت می‌کنیم. اگر سراغ سازمان‌های مختلف برویم، سفره دلشان را باز می‌کنند که مدیر متوجه نیست و... هنر این است که بین مسائل، ABC classification انجام داد و اگر دو مسئله را حل کنیم، بیش‌ترین تأثیر را بر سازمان می‌گذارد. این مسئله، یک مسئله‌ی segmentation است که در حوزه‌های مختلف مطرح است.

پس از این classification با توجه به اهمیت هر بخش، یک CSL برای آن بخش مشخص می‌کنند. مثلاً برای بخش پنج درصدی یک سطح خدمت خوبی را در نظر می‌گیرند.

چرا محصولاتی که کم‌تر اثر دارند را حذف نمی‌کنند تا از شرشان راحت شوند؟ امکان پذیر نیست. چون خیلی از این محصولات ناخواسته هستند. مثلاً در چیپس‌های تولید شده، ممکن است که فقط ده درصد محصولات تولید شده سرعت بالا باشند و پنجاه درصد محصولات تولید شده، سرعت پایین داشته باشند. این موارد در صنعت خیلی رایج است. وقتی صنعت چنین ساختاری دارد، اصطلاحاً Downbining می‌کنند. گفته می‌شود مگر تو سرعت معمولی نمی‌خواستی، من سرعت بالا بهت می‌دهم (به خاطر نداشتن موجودی از محصول سرعت معمولی). مثال این کار این است که شما در بیمارستان نیاز دارید که مثلاً پرستار فشار شما را بگیرد ولی چون پرستارها مشغول‌اند، دکتر این کار را انجام دهد. در خیلی از عملیات، چنین ساختاری وجود دارد.

این چیزی است که ساده به نظر می‌رسد ممکن است یک engine قوی باشد. این ABC classification ممکن است که If then condition های زیادی داشته باشد.

CSL همیشه برای Finished goods مطرح است. کلاً در دنیای بیزینس همیشه Finished goods است که برایش CSL مطرح است.

این که سطح CSL در هر طبقه از محصولات چگونه باشد، به طور شهودی مشخص می‌شود. مثلاً ممکن است که رقبا، ۹۵٪ سطح خدمت در نظر گرفته باشند و ما هم مجبور شویم که به صورت ۹۵٪ مشخص کنیم.

ABC Classification در همه ی حوزه های نظیر مشاوره و HR هم مطرح است. معمولاً از دو بعد Impact سازمانی که چه تأثیری دارد و بعد راحتی بررسی می‌شود (مکنزی). بعد راحتی یعنی این که چه قدر امکان انجام دادن وجود دارد. کارهایی که بیش‌ترین Impact را دارند و راحت‌تر انجام می‌شوند، باید زودتر انجام شوند.

### Impact of Aggregation on safety inventory

وقتی سیگنال‌های مختلف با ارتعاشات متفاوت داشته باشیم و آن‌ها را aggregate کنیم، سیگنال برآیند fluctuation کم‌تری دارد. در واقع با تجمیع یک سری سیگنال، سیگنال aggregate شده، fluctuation کمی دارد. وقتی گفته می‌شود که fluctuation کم می‌شود، یعنی چه؟ نسبت سیگما به میو، بیانگر fluctuation است. از این پدیده خیلی استفاده می‌شود. در دنیای عملیات، اگر یک بافر (با نگاه لوکال) داشته

آمازون، برای کتاب سودآوری کمی دارد اما از موسیقی و اسباب بازی پول خوبی در می‌آورد. چون روی آن‌ها شبکه و وفاداری خوبی ایجاد کرده است.

باشیم، پنج مورد مانند replenishment و.... در این مورد مؤثر است. اگر به جای داشتن موجودی در سه جای مختلف، سیگنال آن‌ها تجمیع می‌شود. در واقع سه بافر، با CSL نود درصد مطرح باشد و با هم تجمیع شوند، مجموع به خاطر این که variation کم می‌شود، میزان موجودی بهینه کم می‌شود.

فرض کنید در سه جا تی شرت می‌فروشید. در سه جای مختلف، تقاضای مختلف می‌آید و fluctuation های خود را دارد. اگر این‌ها را در یک جا تجمیع کنم و مدیریت موجودی را در آن انجام دهم و زودتر ship کنم (مثلاً یک انبار در نقطه‌ای بین این سه شهر)، می‌توان میزان موجودی کمتری را در سیستم انجام داد. شرکت بنتون انواع لباس‌های مختلف در رنگ‌های مختلف را می‌زند. تکنولوژی اش را تغییر داده است. برای این که پدیده‌ی aggregation را انجام دهد، postponement انجام می‌دهد. با این کار توانسته است که موجودی را بهتر مدیریت کند و تغییرات نهایی را در لحظات آخر انجام دهد.

در هولدینگ‌ها هم می‌آیند Cash Flow ها را تجمیع می‌کنند و ریسک نوسانات کاهش می‌یابد.

صنعت بیمه بر اساس demand aggregation است. یک احتمال تصادف وجود دارد. وقتی صد نفر در بیمه عضو می‌شوند، احتمال این که در یک ماه این صد نفر همگی تصادف کنند، خیلی کم است. بعد با توجه به این که هر کدام از این‌ها مقداری پول می‌دهند، می‌توان گفت که صنعت بیمه بر اساس aggregation است.

هر کدام از بانک‌ها می‌بایست مقداری پول نقد داشته باشند. اینکه از هر نوع اسکناس چه مقدار موجودی داشته باشند، مسئله مدیریت موجودی است. حالا اگر ده cashier وجود داشته باشد، یک cashier هم بالاسر این‌ها باشد. برای این cashier، تغییرات تقاضا کم می‌شود. در دنیای inventory management می‌توان CSL را بهبود داد و موجودی را کم کرد.

ممکن است که تجمیع هیچ gain ای نداشته باشد. در واقع aggregarion وقتی است که فرآیندهایی که تجمیع می‌شوند، کاملاً مستقل باشند. خرید در مناطق مختلف، هیچ وقت نمی‌تواند به طور کامل correlated باشد. بیش‌ترین مزایای aggregarion وقتی است که fluctuation بالا باشد. برای بنتون اگر fluctution کم باشد، ارزش سرمایه‌گذاری در آن حوزه را نداشت ولی postponement برایش خیلی مفید است.

اگر سیگنال‌های مختلف وجود داشته باشد و یکی از آن‌ها خیلی غالب باشد، تجمیع و aggregate کردن آن، شبیه سیگنال غالب می‌شود و aggregation مفید نیست.

وقتی اتاق اورژانس بیمارستان را برونسپاری می‌کنند: در اورژانس عده‌ای باید سریعاً به آن‌ها پاسخ داده شود. اما ممکن است که در طول شبانه روز، دو نفر این طوری مراجعه کنند. لذا utilization بسیار پایین است ولی آدم‌ها بسیار گران می‌باشند. چون هزینه staff بالاست، utilization کم است. خیلی از بیمارستان‌های خارج از کشور بر اساس demand aggregation برونسپاری کرده‌اند. وقتی یک دکتر به صورت پارت تایم در جایی کار کند، نرخ کارش بسیار بالا می‌رود. مثلاً ریت حقوقش از ساعتی ۱۰۰ دلار به ۸۰۰ دلار می‌رسد. بنابراین همه

پس وقتی که سیگنال‌ها  
correlated باشند،  
aggregation مفید نیست.

به خاطر demand  
aggregation است که VMI  
برای چندسپلایر به درد  
نمی‌خورد.



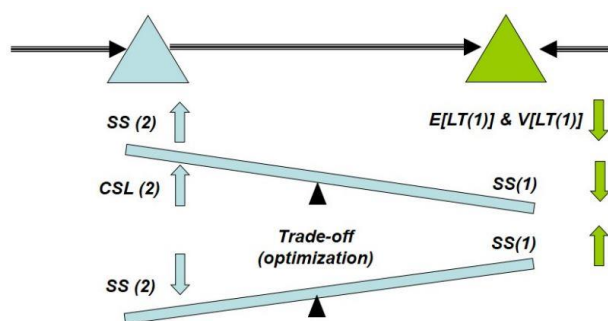
دوست دارند که پارت تایم کار کنند. اشکال این جاست که اگر مریض از او شکایت کند، این قدر برای دکتر گران تمام می‌شود که کسی تمایل به کار پارت تایم ندارد. بیزینس‌هایی که برای اورژانس تشکیل شده، کارش همین است. برنامه‌ریزی می‌کند. مثلاً شما خانه تان این جاست، در این ساعت on-call باش که اگر نیاز شد، کم‌تر از پنج دقیقه در مکان حضور داشته باشی و ساعتی مثلاً هفتصد دلار به تو می‌دهم. یک تیم وکلا در این جا وجود دارد که تمام کیس‌های حقوقی را هندل می‌کنند. با این کار هزینه بیمارستان‌ها کم می‌شود، درآمد پزشک‌ها و جراح‌ها خیلی بالا می‌رود و آن شرکت‌ها هم سود می‌کنند. این شرکت‌ها گاهی با فرودگاه‌ها قرار دارند، با هلی کوپتر مثلاً اگر تماس گرفته شد، عرض سه دقیقه دکتر را به استان بغلی ببرید. وکیل‌ها برای دکترها، economies of scale دارند. دکتر در حالت استخدام رسمی، کلی هزینه نظیر bonus و... دارد و باید total cost of ownership را در نظر گرفت.

در نگاه گلوبال، بسته به این که ما چه قدر می‌خواهیم responsive باشیم که موجودی finished goods باشد، مدیریت موجودی می‌تواند تا بالادست زنجیره تأمین مؤثر باشد یا اینکه در ابتدای کار تمام باشد.

هر چه به سمت بالا دست برویم، flexibility زیادتر می‌شود ولی مدیریت موجودی در پایین دست، با responsive چالش دارد.

### multii-echelon

هر چه در بالادست، موجودی بیش‌تری را نگه داریم، مانند این است که CSL این نقطه زیادتر می‌شود، یعنی درصد مواردی که از بالادست تأمین می‌شود زیادتر است. هر چه CSL اینجا زیاد شود، وابستگی پایین دست به بالادست کم‌تر می‌شود و fluctuation rate کم می‌شود. هر چه وابستگی به بالا دست کم شود average lead time کم می‌شود و safety stock کم‌تر می‌شود. می‌توان به طور گلوبالی بالانس کرد که چه ترکیبی از CSL برای هر بخش از زنجیره تأمین مفید است. یعنی گفته می‌شود که این بستگی به چیدمان موجودی دارد. خیلی وقت‌ها در این صورت می‌توانم موجودی کم‌تری داشته باشیم و CSL هم زیاد شود. این چیزی است که امروزه در صنعت به شدت مطرح است. ممکن است به جای دو استیج، چندین استیج وجود داشته باشد. با روش الگوریتم‌های هوشمند، می‌توان چیدمان را به گونه‌ای پیدا کرد که موجودی به بهترین شکل مدیریت شود.



در صنعت Semiconductor، ۲ استیج اصلی داریم. Finished Goods، خروجی محصول و Semi Finished که مواد اولیه است. با اجرای Multi-echelon، چند ده درصد سطح موجودی را کاهش دادند و CSL را حفظ کردند. در خود کارخانجات تولیدی، جای inventory را کاهش دادند و سود زیادی کردند.

چیزی که امروزه در صنعت به خصوص در فضای retail مهم است assortment است. یعنی من فقط نباید موجودی را بهینه کنم، بلکه به محصولاتی که correlation دارند (مثل شامپو بچه و پوشک) باید توجه کرد. ترکیب محصولات باید به خوبی مدیریت کرد.

اگر بخواهیم مالتی - اشلون یا گلوبال اینونتتری را ببینیم، به نتورک استراکچر هم کار داریم. خیلی وقتی‌ها نیاز به inventory cost هم داریم. هزینه‌ی inventory در finished goods با هزینه inventory در مواد خام فرق دارد.

### objective function

معمولاً کم کردن total inventory cost هستیم. اگر تفاوت هزینه‌ی تمام شده بین finished goods و پایین دست و بالادست خیلی فرق نداشته باشد، می‌توان برای سادگی مسئله به total inventory cost برویم.

وقتی به صورت گلوبال بهینه سازی برای موجودی انجام می‌دهیم، می‌گوییم که CSL می‌خواهیم چگونه باشد و موجودی در کل اینونتتری چه چیدمانی داشته باشد تا CSL که مد نظر است را بش برسیم. هزینه‌ی موجودی ممکن است این قدر زیاد باشد که شرکت‌ها حاضر نباشند این قدر هزینه کنند، لذا در دنیای عمل به budget optimization روی می‌آورند. مثلاً اگر هزینه‌ی ۱۰۰ میلیون لازم است و ۴۰ میلیون بیش تر بودجه نداریم. اینجا مسئله مهندسی معکوس رخ می‌دهد و گفته می‌شود که من با ۴۰ میلیون بهترین سطح CSL ای که می‌توانم داشته باشم، چه قدر است؟ در واقع گفته می‌شود که با این هزینه و رسیدن به بهترین سطح CSL، بهترین network structure چگونه است؟ برای رسیدن به این سطح CSL و این که بودجه بیش تر از ۴۰ میلیون نشود، بهترین چیدمان چگونه است؟

### Typical Inventory Process

قبلاً نگاه Buffer Centric و locality وجود داشت. موجودی همیشه به دو صورت بیان می‌شود: ۱-unit یا ۲- بر اساس replenishment بالا دست. در واقع گفته می‌شود days of cover چیست؟ مثلاً من باید به اندازه‌ی ده روز باید کاور داشته باشم. بنابراین بر اساس این که cycle time چه قدر باشد، یک ضربی هم در آن ضرب می‌شود. بنابراین بر فرض تقاضا که چه demand به مارکت منتقل می‌شود، بررسی می‌شود. inventory target یک بار مالی دارد و مدیریت باید approve کند و نمی‌توان بدون دلیل مشخص کرد. سپس این targetها بررسی می‌شوند.

اینگونه نیست که صنعت با یک شات بگوید که بهترین سطح موجودی چه قدر است؟ در واقع همیشه تمام حالت‌های Minimize total inventory و cost و Minimize total inventory و Budget و optimization بررسی می‌شود.

زمانی که در ابتدای لانچ کردن محصول هستیم و responsive بودن بالاست، باید به بودجه بیش تر اهمیت کرد.

این سیستم سنتی و خیلی بر اساس average ها است و واریانس‌ها خیلی لحاظ نمی‌شود. در حالی که اگر variability پایین دست و بالا دست حساب کنیم.

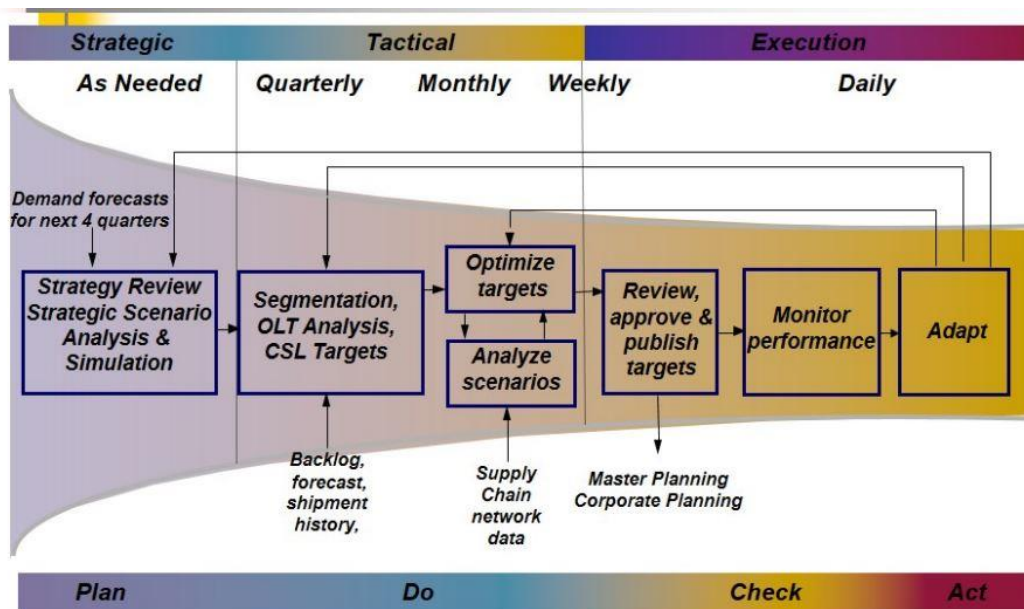
این پروسه در گذشته بسیار manual بوده است و بسیار زمان‌بر بوده است. مدیریت موجودی مانند مدیریت خون در رگ‌های بدن است. مدیریت موجودی قبلاً به صورت ad hoc بوده است و روند خاصی نداشته است.

لزوماً برای مدیریت موجودی به صورت global الگوریتم‌های پیچیده لازم نیست. اگر advanced planning system ها باشد، شرایط بهتر می‌شود. ولی اگر نباشد می‌توان از الگوریتم و حلقه‌ی PDCA استفاده کرد.

حلقه‌ی PDCA: اگر سیستم، close loop باشد، همیشه می‌تواند خودش را اصلاح کند. وقتی سیستم حلقه بسته است، خودش Learning انجام می‌دهد و مشکلات سریع شناسایی می‌شوند. وقتی سیستم یادگیرنده باشد، به سادگی می‌توان آن را کنترل کرد. fuzzy logic در واقع، می‌آید یک سیستم پیچیده را به روش ساده‌تری کنترل می‌کند. آنچه مهم است، close loop بودن است.

### Closed Loop Process in Practice

همیشه سیستم برنامه‌ریزی یک حالت قیف مانند دارد:



آنچه در حوزه‌ی مدیریت موجودی وجود دارد، سه حوزه strategic و tactical و execution دارد. هر چه به زمان حال و execution نزدیک می‌شویم، دقت باید بسیار بالا رود. در حالت استراتژیک، دقت کم‌تر می‌شود.

استراتژی چیزی است که در کل life cycle سایه می‌اندازد. در کل life cycle فضا رقابتی است و استراتژی این گونه نیست که عوض شود. در سر دیگر طیف، فضا روزانه عوض می‌شود. صبح ممکن است که یک جور باشد و شب جور دیگر باشد. در حالت tactical نه مانند execution روزانه و ساعتی است و به لختی استراتژیک هم

نیست. سیاست‌هایی که مابین این دو است. در این حالت تصمیم‌گیری‌ها بیشتر برای یک کوآرتر است. در حالت تاکتیکی ممکن است که بخواهیم مثلاً در یک فصل، ۲ محصول را با هم باندل کنیم. تاکتیک‌هایی که در میان مدت وجود دارد، به کار گرفته می‌شود.

## مدیریت تولید

در بعضی از مدرسه‌های کسب و کار، این یک درس جدایی است. در واقع تمام Procurement و موجودی به خاطر گشتن چرخ تولید است.

جنبه‌های مختلف می‌تواند داشته باشد. جنبه‌های مدیریتی و جنبه‌های ریاضی برای مدیریت تولید وجود دارد. شاید یک جنبه این باشد که planning چه قدر پیچیده است و لازم است که بدانیم که جغرافیایی دارد و این که چه چیزهایی را نمی‌دانیم، هدف خوبی برای بررسی مدیریت تولید است.

کسی خوب می‌تواند production management را به خوبی انجام دهد که بتواند دنیای واقعی را represent کند. رویکردهای managerial بررسی می‌شود مدل‌های سنتی مانند MRP گفته می‌شود.

سیستم‌ها، دو بعد conflicting از نظر تولید دارند: بعد volume و بعد variety. نمی‌توان هر دوی این‌ها را به صورت زیاد داشت. لذا کلاً همیشه با دو تردی‌آف بین این دو مطرح است و باید دید که پوزیشن چه می‌تواند باشد.

Globalization می‌تواند هم فرصت باشد و هم تهدید. در حالت فرصت می‌توان، از کارگران در جاهای ارزان‌تر استفاده کرد. از نظر تهدید این است که می‌تواند یک فضای رقابتی شدیدی را ایجاد کند. یادمان باشد که کنار low cost، امروزه responsive بودن به خاطر انتظارات بازار و تنوع‌های موجود خیلی مهم است. می‌توان با توجه به نظریه داروین گفت که الزاماً قوی‌ترها امکان ماندن بیشتری ندارند. کسانی می‌مانند که چابک‌تر باشند و با محیط سازگار باشند و responsive بودن باشند. این خیلی مهم است که چه کسی محصولش را به بازار ارائه کند. هر چه قدر هم low cost بود و به انتظارات بازار و responsive بودن توجه نکرد، احتمال موفق شدن کم است.

تکنولوژی‌هایی نظیر process technology و information technology فرآیندهای تولید را تحت تأثیر قرار داده‌اند. پارادایم انتخاب volume و variety وجود دارد که این‌ها متأثر از سه نوع فعالیت است. develop کنیم، تولید کنیم و بفروشیم.

خیلی از شرکت‌ها زمین می‌خورند به این خاطر این که واحدهای فروش و تولید با هم رابطه‌اشان ضعیف است.

در فرآیندهای NPD، کسی که design می‌کند نمی‌تواند از product فاصله بگیرد و بر عکس

در دوران گذشته روند NPD خیلی sequential بودند ولی اکنون به خاطر فضای global این طور است که در هر فرآیند، گفته می‌شود که اگر مثلاً یک مرحله‌ی خاص انجام شد، مرحله در یک فرآیند دیگر می‌تواند شروع کند. در واقع فرآیندها به هم تنیده شده‌اند.

حوزه‌ی manufacturing یا production این مراحل را طی کرده است:

craft production: تقاضا به صورت خاص بوده است و تولید به صورت کارگاهی.

mass production: به جای این که یک آدم یا شرکت را تارگت کنیم، کل مارکت را تارگت می‌کنیم.

mass customization: روی مارکت و سگمنت‌های مختلف تمرکز داریم نه روی خود محصول.

presonalized production: من برای این آدم خاص یا شرکت خاص، چه محصولی را عرضه کنیم. تفاوت زیادی بین این‌ها وجود دارد و اینکه ویژگی هر کدام از این حوزه‌ها چگونه است، مطرح است.

در mass production، تنوع کم است و حجم بالاست.

Personalized به صورت cost-effective و زمان مناسب شکل می‌گیرد.

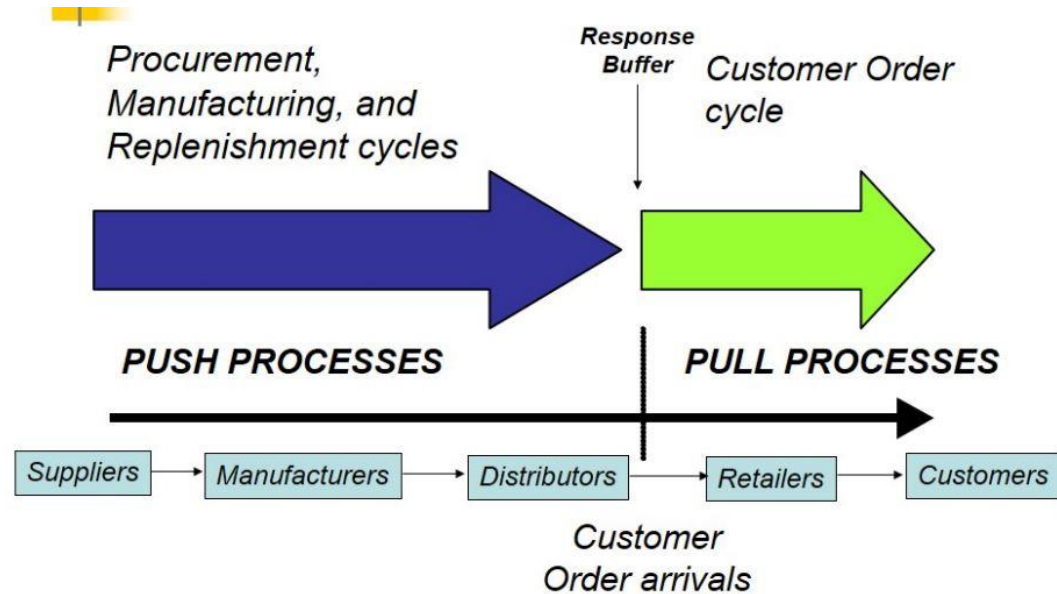
response buffer، اگر در زنجیره تأمین مراحل مختلفی داشته باشیم، زنجیره تأمین بستگی به این دارد که ما تا کدام مرحله می‌خواهیم جلو برویم. یعنی تا جایی تولید را ادامه می‌دهیم ولی در یک مرحله، تا سفارش نیاید، تولیدی انجام نمی‌شود. اگر response buffer را به آخر زنجیره تأمین ببریم، بحث forecast وجود دارد. این حالت make to stock یعنی تا finished good را می‌دانیم و بر اساس آن تولید می‌کند. حالت دیگر response buffer می‌تواند روی مواد اولیه باشد. یعنی تا سفارشی نیاید، چیزی تولید نمی‌کند و این حالت make to order می‌باشد. در زنجیره‌ی تأمین، Stage‌های مختلفی وجود دارد. تا یک مرحله بر اساس forecast جلو می‌رویم و بیش‌تر از آن تولید نمی‌کنیم. مگر این که مشتری سفارشی بدهد و ما بدانیم قطعی است.

در صنعت کامپیوتر شرکتی صد در صد make to stock عمل می‌کند و شرکت دیگری make to order عمل می‌کند و هر دو موفق می‌باشند. در واقع این شرکت دو سگمنت مختلف را تارگت کرده‌اند. مثلاً دل کامپیوتر مشتریانی را تارگت کرده است که کامپیوتری با ویژگی‌های خاص را می‌خواهند و می‌توانند صبر کنند. معمولاً اگر زمان بعد از سفارش تا تولید زیاد باشد، ممکن است کار درستی نباشد. بستگی به variety سیستم، این‌ها می‌تواند متفاوت باشد.

همیشه second mover استراتژی low cost دارد. این هم در عمل اثبات شده است و هم در تئوری.

یک exterem این response buffer در حالت make to stock است که ابتدا تولید می‌شود و سپس توسط مشتری خریداری می‌شود و حالت دیگر این است که make to order باشد، که چیزی تولید نمی‌شود تا سفارشی داده شود.

Push processes مثلاً در صنعت خودرو تمام بدنه و شاسی را تولید کرده‌ام. این حالت صد در صد بر اساس forecast می‌باشد. ولی رنگ کردن خودرو و... را تا سفارش نیامده است، انجام نمی‌دهم. وقتی بر اساس سفارش عمل می‌کنم، حالت Pull processes می‌شود.



اگر سیستمی داشته باشیم که صد در صد بر اساس make to order باشد، آیا نیازی به forecasting هست؟  
 من اگر صد در صد هم make to order باشم، بالاخره raw material را که نیاز دارم. مثلاً دل کامپیوتر اگر بخواهد، برای مادربرد و ... هم make to order عمل کند و وقتی سفارش آمد، آن‌ها را بخرد. lead time آن به شدت بالا می‌رود. پس اگر صد در صد هم make to order باشد، برای raw material نیاز به forecasting وجود دارد. در این حالت، اگر بتوانم موجودی را aggregate کنم، هزینه‌ی موجودی پایین می‌آید.

### Product Variety

هر چه قدر variety بالا رود، (ظرفیت تولید محدود است) باید ببینم که با این ظرفیت محدود، از هر محصول چه قدر می‌توانم تولید کنم و تقاضا disaggregate می‌شود و uncertainty بالا می‌رود. aggregation موجودی گفته شد برای یک محصول بود. بعد زمان هم می‌تواند مطرح باشد. اگر بخواهم برای سال آینده forecast کنم، که چه قدر یک محصول فروش می‌رود، خطا یک مقداری است. وقتی به جای سال، فصلی، ماهیانه و هفتگی بخواهید forecast کنید، خطای پیش‌بینی بالا می‌رود. حتی aggregation در بعد زمان هم می‌تواند مطرح باشد.

هر قدر تقاضا disaggregate شود، دقت forecast کم می‌شود. یعنی انحراف معیار بالا می‌رود و این یعنی برای یک میزان CSL باید موجودی بیشتری داشته باشم و این یعنی هزینه‌ی بالاتر.

وقتی MTO عمل می‌شود که از بالادست تا پایین دست زنجیره تأمین، سریع طی شود. در غیر این صورت، MTS باید عمل کرد.

اگر شرکتی محصولی دارد که می‌داند هر چه قدر تولید کند، بازار می‌خرد، آن را make to stock تولید می‌کند ولی اگر ماشینی را مثلاً تولید می‌کند که هر کسی نمی‌خرد، make to order عمل می‌کند.

هزینه که بالا می‌رود، یک tradeoff وجود دارد، چون ممکن است که یک سری مشتریان بروند و یک سری مشتریان بیایند. برای این که بازار را از دست ندهم، باید یک میزان variety داشته باشم و همیشه بین میزان variety و هزینه‌ای که متقبل می‌شوم یک tradeoff وجود دارد. مثلاً آدیداس وقتی میزان تنوع را بالا برد، (اسکن کردن پا) نتوانست به خوبی جواب دهد.

## Market and Society needs

انواع خط تولید وجود دارد.

### Dedicated Manufacturing lines

صرفاً برای تولید یک محصول است. مثلاً خط تولید پراید در سایپا، فقط و فقط پراید تولید می‌تواند بکند در صورتی که خط تولید ۲۰۶، انعطاف پذیر است.

Regionalization و Personalized production به نوعی مشخص می‌کند که مثلاً برای هندی‌ها باید ماشین ارزان و برای عرب‌ها ماشین بزرگ تولید کرد.

## Planning

اگر بخواهیم، خوب برنامه‌ریزی انجام دهیم باید دنیای واقعی را به خوبی مدل کنیم.

Building Block ها در مدل کردن دنیای واقعی به کار برده می‌شوند.

مفهوم Item: هر متریالی، پارت و کامپوننت

Buffer: آیتمی که در یک لوکیشن باشد. در دنیای واقعی هیچ وقت آیتم نداریم، Buffer داریم. در واقع Item همیشه در لوکیشن است.

Operation: انواع عملیاتی که ممکن است رخ دهد، مثلاً ممکن است که آیتم را به آیتم دیگری تبدیل کند یا این که آن را از نقطه‌ای به نقطه‌ای دیگر ببرد.

Flow: هر بافری به هر عملیاتی از طریق flow وصل می‌شود.

Resource: هر عملیاتی نیاز به resource دارد. perishable می‌باشد. resource مفهوم consumption دارد و اگر در زمان مد نظر استفاده نشود، perishable است.

هیچ وقت نمی‌توان یک بافر را به بافر دیگری وصل کرد و حتماً بین آن‌ها یک عملیات وجود دارد.

انواع operation:

Alternate operation: مثلاً می‌توانم از طریق هوایی و یا از طریق زمینی و کامیون، حمل و نقل را انجام دهم. باید مشخص شود که Primary چیست و سپس بر اساس اولویت، عملیات alternative مشخص شود. اگر این

در سیر تاریخی، انواع رویکردهای تولید، ابتدا تنوع زیاد بوده است (craft)، سپس کم می‌شود (mass production) و سپس دوباره کم می‌شود (mass customization).

building blockها به درستی مشخص نشوند، نمی‌توان دنیای واقعی را به خوبی مدل کرد. بر خلاف primary alternate یک حالت proportional وجود دارد. در این حالت، مثلاً گفته می‌شود که شصت درصد باتری مورد نیاز، از یک باتری ساز و مابقی از دیگری تأمین می‌شود.

Rounting operation: وقتی یک کاری شروع می‌شود، چند عملیات شکل می‌گیرد که بین آن‌ها مجاز نیستیم که Inventory داشته باشیم. مثلاً خط تولید خودرو یا ویفر نیمه رساناها این طور است. خط که شروع می‌شود باید بدون توقف بروند. چرا این کار را به عنوان single operation تعریف نمی‌کنیم؟ چون هر step ممکن است که resource خودش را داشته باشد که با بعضی share شود و با بعضی از منابع share نشود.

انواع Flow: برای flow هم همین حالات وجود دارد:

مثلاً Alternate flows (Or Condition) که معمولاً دو فلش پر نمی‌شود (یعنی روی هم قرار نمی‌گیرند) و حالت دیگر که برای مونتاژ است حالت Simultaneous flow می‌باشد (در این حالت دو فلش روی هم قرار می‌گیرند).

حالت Co-Product (binning operation) در واقع بعد از یک عملیات، ممکن است که محصولات با ویژگی‌های مختلف تولید شود. یعنی آیتم‌های مختلف به جود بیاید. این ساختاری است که diverging flows می‌باشد. هر جا test operation وجود دارد، این حالت می‌تواند رخ دهد.

مفهوم alternate resource هم وجود دارد. مثلاً اگر حالت حمل بود، راننده می‌تواند ممدآقا باشد، می‌تواند حسن آقا باشد.

این Building blockها می‌توانند کنار هم باشند و دنیای واقعی را مدل کنند.

برای هر بافر، بررسی می‌شود که چه قدر محصول on-hand می‌باشد. برای مدل شدن این که چه برنامه ای در یک بافر دریافت می‌شود، یک schedule وجود دارد.

برای عملیات، cycle time، و setup time یا Cooling time وجود دارد. setup و cooling یکی قبل از شروع عملیات و دیگری پس از تمام شدن عملیات است.

WIP محصولی است که در عملیات است. در واقع تعداد قبل و بعد در بافرهای عملیات، تعداد یکسان محصول وجود ندارد، آنچه کم است WIP است.

برای Flow، این پارامترها مطرح است:

برای resource هم مفهوم calender وجود دارد که برای یک بازه‌ی زمانی چه ظرفیتی وجود دارد. مثلاً هشت نفر در روز.



این‌ها پارامترهای استاتیک بوده است.

پارامترهای دینامیک:

برای یک بافر می‌توان تقاضا را تعریف کرد. تقاضا یا از جنس forecast یا order می‌باشد. order همیشه point in time است. من ساعت دو باید این تعداد محصول را تحویل مشتری دهم. forecast مفهوم bucketized است. مثلاً برای خرداد این قدر پیش‌بینی دارم. این دو nature مختلف، planning را پیچیده می‌کند.

این‌ها هر لحظه ممکن است که تغییر کند و به زبان فنی state سیستم ممکن است تغییر کند. میزان-on hand ممکن است تغییر کند. میزان WIP، ظرفیت و حتی supply calender می‌تواند، به طور مداوم تغییر کند. دیده می‌شود که چه تعداد زیادی پارامتر می‌تواند تغییر کنند.

خروجی‌ها: بعد از اینکه planning انجام می‌شود، باید دید که این فورکست، برایش چه اتفاقی افتاده است. یادمان باشد که order، تعهد دارد. این که چه مقدار را تحویل می‌دهیم، خیلی وقت‌ها می‌تواند backlog شود. در واقع سفارش می‌تواند به حالت پس‌افت برسد و با تأخیر تحویل دهیم ولی برای پیش‌بینی (forecast)، backlog مفهومی ندارد.

Operation plans: می‌بینیم که چه زمان‌هایی چه کاری روی عملیات انجام می‌شود. بعد از planning می‌توانیم ببینیم utilization این برنامه از یک منبع چگونه است؟

در نیمه رساناها سفارش‌ها ممکن است از سه ماه تا چندین ماه طول بکشد. سفارش ممکن است برای نه ماه آینده باشد. چون سیکل fabrication خودش سه ماه طول می‌کشد. در تگزاس اینسترومنت، ۳۴ هزار تا Chips تولید می‌شد. یعنی ۳۴ هزار عملیات انجام می‌شد.

یک قانون سر انگشتی این است که چه قدر در آینده، سفارش دریافت می‌کنیم. از مواد اولیه تا finished goods باید دو برابر آن را در نظر گرفت.

وقتی یک منبع، بین چندین عملیات shared شود. محاسبات optimization بسیار پیچیده می‌شود.

### Some planning guidelines

هر آنچه که ممکن است، bottle neck و گلوگاه شود، را ناچاریم که مدل کنیم. به عنوان مثال در سیستم حمل و نقل، اگر تعداد تراک‌ها محدود است، آن‌ها را مدل می‌کنیم. اگر تعداد راننده‌ها محدودیت باشد، آن‌ها را مدل می‌کنیم. خطوط هوایی، برایشان خلبان حتماً یک bottle neck است. در نتیجه می‌بایست مدل شود. مثال دیگر وقتی است که می‌خواهیم صنعت خودرو را مدل کنیم. در سیستم برنامه‌ریزی خودرو هیچ وقت پیچ

planning time bucket  
یعنی این که برنامه ریزی  
انجام شده، به صورت ماهیانه  
است، هفتگی است یا روزانه  
است؟

هر چه به زمان حال نزدیک  
می شود چون به  
excecution نزدیک می‌شود،  
پس باید time bucket ها  
ریزتر شود.

و مهره را جزء بافرها نمی‌گذارند. پیش‌فرض این است که پیچ و مهره هیچ گاه، bottle neck نمی‌باشد. در صنعت نفت و گاز، بر خلاف صنعت خودرو، پیچ و مهره می‌تواند bottle neck باشد.

سوال این است که planning horizon چگونه باشد؟ دو هفته باشد؟ دو ماه باشد؟ یک سال باشد؟ چه قدر باشد؟ خیلی وقت‌ها در صنایع مختلف این بستگی به میزان سفارشی است که دریافت می‌کنیم. وقتی که lead time عملیات زیاد است، معمولاً سفارش‌ها برای زمان‌های دورتر است. مثلاً در صنعت نیمه‌رساناها سفارش به خاطر این که عملیات ممکن است سه ماه طول بکشد، ممکن است تا نه ماه طول بکشد. معمولاً بر اساس این که سفارش چه قدر در آینده است، از مواد خام تا محصول نهایی را می‌بینند چه قدر است و دو برابر آن را در نظر می‌گیرند. یعنی از scratch تا finishing goods که برنامه ریزی کردیم، یک سیکل دیگر هم بتوانیم پاسخ‌گو باشیم. دل کامپیوتر معمولاً سیکل برنامه‌ریزی‌اش دو ماه تا سه ماه است. گرچه عملیات مونتاژ کامپیوتر تکنیکالی در حد چند ساعت و در عمل یک هفته است ولی برنامه ریزی را تا دو ماه انجام می‌دهد.

ظرفیت چگونه باید مدل شود؟ مثل time bucket می‌ماند. هر چه به زمان حال نزدیک تر می‌شویم، گرانولاریتی باید بیش‌تر شود و دقت برنامه‌ریزی باید بالاتر برود.

### ERP (Enterprise Resource Planning) <--> APS (Advanced Planning System)

حوزه‌ای که از زمان حال به گذشته است، در حوزه‌ی transaction system ها یا ERP می‌باشد. همیشه گذشته دیده می‌شد و می‌گفتند که state آینده باید چگونه باشد. به طور سنتی ERP که state گذشته را دارند، APS ها بر روی آن‌ها نشسته‌اند. بعدها، شرکت‌هایی ERP خودشان ماژول‌های planning را توسعه دادند. APS ها هم، کمی به فضای data manager وارد شدند. در APS ها گفته می‌شود که در هر لحظه از آینده می‌خواهیم state ما چگونه باشد.

در planning یک مفهوم قیف مطرح است. هر چه به زمان حال نزدیک می‌شویم، برنامه ریزی باید دقیق شود. در زمان حال باید بدانیم که دو دقیقه‌ی دیگر چی کار می‌خواهیم بکنیم. وقتی به زمان حال نزدیک می‌شویم، time bucket روزانه است و هر چه عقب‌تر برویم، time bucket ها ممکن است تا فصلی هم برسد. مثلاً باکت‌سایز دل کامپیوتر به ماه نمی‌رسد. ولی Intel، planning horizon یک ساله دارد. اواخرش ماهانه است. سه هفته‌ی اولش روزانه است.

### پیچیدگی‌های حوزه‌ی برنامه‌ریزی تولید

**Continuous v. Discrete time domain:** دو نوع تقاضا داریم، یکی از نوع order می‌باشد که continuous است و یک نوع forecast داریم، که bucketize می‌باشد و discrete است و گفته می‌شود که در هر bucket پیش بینی من، مثلاً دویست است. forecast مثلاً برای هفته‌ی یا ماه آینده داریم ولی order برای یک زمان مشخص است. لذا order می‌تواند با کمیت‌های مختلف برای زمان‌های مختلف باشد ولی forecast، bucketize می‌باشد و گفته می‌شود که پیش‌بینی من برای فلان bucket پانصد است. برای سفارش، همیشه

یک زمانی زودتر، تولید شکل گرفته می‌شود و در واقع این همان مفهوم JIT است (در این حالت، با در نظر گرفتن yield، گفته می‌شود که برای یک هفته، یک میزان اگر تولید کنم، کافی است) ولی برای دنیای forecast متفاوت است. یعنی بیهوده زودتر برنامه‌ریزی نکنم. اگر نیاز نیست که زودتر از یک هفته برنامه‌ریزی کنم، این کار را انجام ندهم. Forecast این‌گونه نیست. چون backlog ندارد و برای یک time bucket است. سوال این است که چه زمانی برنامه‌ریزی کنم؟ در دنیای صنعت رویکردهای بسیار متفاوتی وجود دارد. مثلاً فرض کنید یک bucket ماهیانه وجود دارد و این متشکل از چهار هفته است و forecast هزار عدد می‌باشد. مثلاً برای این نوع T-shirt، برای خردادماه پیش‌بینی می‌شود که هزار عدد نیاز است. رویکردهای متفاوت در صنعت وجود دارد. یک گزینه این است که بعضی از شرکت‌ها می‌گویند که باید کمی conservative باشیم. درست است که forecast برای کل یک ماه است ولی من برای روز اول ماه، موجودی بیش‌تری آماده می‌کنم و از همان روز اول، برای پیش‌بینی آماده‌ام. بعضی از شرکت‌ها معتقدند که این کار ما را از فضای JIT دور می‌کند و خوب است که even distribution داشته باشیم. برای هر هفته از چهار هفته‌ی ماه، ۲۵۰ عدد داریم. شرکت‌ها نسبت به سفارش، تعهد دارند ولی در forecast هیچ commitmentی نسبت به بازار وجود دارد.

این مفاهیم، عیناً در حوزه مالی و حوزه بودجه، مطرح است. یا باید در تاریخ خاصی تحویل داده شود یا هزینه‌ی forecast وجود دارد. مفاهیم را اگر بدانید، در جاهای دیگر، می‌توانید ایده بگیرید که چه کار کنید.

فرض کنید در ماه خرداد، هزار عدد پیش‌بینی دارم و هیچ سفارشی وجود ندارد، ولی در زمانی که می‌خواهم plan کنم، ۱۰۰ سفارش شکل می‌گیرد. خالص پیش‌بینی (neted forecast) ۹۰۰ عدد می‌شود و برای این عدد برنامه‌ریزی می‌شود.

اثر شلاقی، در کسب‌وکارهای B2B مؤثر است. هر چه به سمت بالادست زنجیره تأمین می‌رویم، اثر شلاقی بیش‌تر دیده می‌شود. هر چه leadtime طولانی‌تر شود، باید نسبت به replenishment time، کنترل داشته باشیم و میزان موجودی را بالا می‌برم. information technology کمک می‌کند که این‌ها، بهتر شود. ولی علی‌رغم این، خیلی وقت‌ها کسی دوست ندارد که اطلاعات به بالا دست برود. شرکت‌های B2B باید برای موج‌های اثر شلاقی آماده باشند و باید به خوبی forecast کنند و خودشان را پروتکت کنند. مثلاً شرکت اینتل، مایکروپروسسور می‌زند، کامپیوتر نمی‌زند ولی برای سیستم planning نیاز به forecasting دارد.

زنجیره‌ی تأمین اگر یک سری استیج‌ها باشد، هر چه به سمت بالا دست می‌رویم، aggregation انجام می‌شود. یعنی ممکن است که صدها نوع محصول مثل مادربرد و... وجود داشته باشد که در جاهای مختلف مصرف می‌شود. بنابراین این صداها به ۱۰ آیتم، map می‌شود و این aggregation منجر به کاهش fluctuation می‌شود. ما به طور لوکالی، demand aggregation را داریم، اما globally که نگاه کنیم، هر شرکت می‌خواهد نسبت به بالادست protect باشد و به موقع پاسخ پایین دست را بدهد. در نتیجه خودش amplified می‌کند. accross enterprise، ما amplification داریم. این که superposition این amplificationها چگونه

است، بستگی به صنعت دارد. بعضی از صنایع که کل life cycle کوتاه است، aggregation مهم است. ولی در صناعی که lead time زیاد، اثر شلاقی وزن زیادتری دارد.

### Demand priorities

منابع ما محدود است و برای این که منابع ما محدود است باید در مقام مصرف منابع محدود، کمی باید رعایت اولویت را داشته باشیم. برای این که از منبع محدود به طور بهینه استفاده کنم، باید اولویت بندی کنیم. این اولویت بندی خودش یک مسئله‌ی مهم و چند بعدی است. سه بعد product و time و customer مطرح است و باید اولویت بندی کرد که کدام product مثلاً مهم‌تر است. آیا ابعاد دیگری وجود ندارد؟ بله، مثلاً مارجین سود، بسیار مهم است.

مشتریان، انواع مختلفی دارند. محصولات هم، انواع مختلفی دارد. این ها عملاً یک بعد نیستند یک hierarchy هستند. مشتریان یک سلسله مراتب هستند:

مشتریان

مشتریان بین الملل - غیر بین الملل

محصول هم همین طور است:

product line 1, product line 2

product family 1, product family 2

لذا وقتی صحبت از تقاضا می‌کنیم، سه hierarchy در هم ضرب می‌شوند و feasible space را مشخص می‌کند. اینکه چگونه در فضای feasible space این‌ها، تلاقی داشته باشیم، واردش نمی‌شویم و برای این سه Hierarchy باید، forecast کنیم.

### Dependent vs. Independent demand

خیلی از تقاضاها عملاً از سمت مشتری می‌آید و این تقاضاها Independent است و دست من نیست. چون مشتری خودش تصمیم می‌گیرد که بخرد یا نخرد.

این که من چه قدر می‌خواهم زودتر یک هفته را پلن کنم، dependent خود من هست. این که من بالا دست می‌خواهم چی کار کنم، بستگی به خودم دارد. market side به من دیکته می‌کند ولی این که من چه عکس العملی به آن نشان دهم، مهم است. ممکن است که من چون محدودیت ظرفیت دارم و می‌خواهم سه روز هم زودتر آماده باشم، گونه‌ای که دوست دارم، عمل می‌کنم.

در صنایع که leadtime کم هست، fluctuation زیاد است ولی در صناعی که leadtime زیاد است، demand aggregation رخ می‌دهد.

به عنوان قانون سرانگشتی، هر چه به زمان حال نزدیک‌تر می‌شویم، اولویت بیش‌تر می‌شود.

**Date effective**

آنچه دنیای واقعی می گوید، این است که تک تک پارامترهایی که برای هر building block در نظر گرفته شده است، time dependent است. هر چه زمان طولانی تر باشد، این حتماً باید لحاظ شود. همه‌ی operation ها، time dependent است.

**Frozen window**

بسیار چیز رایجی هست. شرکت‌هایی که در فضای رقابتی کار می‌کنند نه فضای مونوپولی، مجبور هستند که این موضوع را رعایت کنند. مثلاً داریم برای time window سه ماهه، Plan می‌کنیم. فرض کنید، هر روز این را تکرار می‌کنیم. هر روز می‌گوییم از امروز به بعد، می‌خواهیم state system آینده را تعیین کنیم. یک روز که گذشت برنامه‌ریزی می‌کنم، امروز باید چی کار کنم. اگر سیستم هر روز تغییر جهت دهد، سیستم chaotic می‌شود. مثلاً یک ماشین را امروز بگویم بفرستید اصفهان، فردا بگویم بفرستید یزد. چنین شرایطی هرج و مرج می‌شود. برای این که شرکت‌ها کمی فضا را stable کنند، مفهوم frozen window را مطرح می‌کنند.

برنامه‌ریزی به طور سه ماهه است. من امروز تا سه ماه دیگر را برنامه‌ریزی می‌کنم. یک روز که گذشت، دوباره از آن روز تا سه ماه دیگر برنامه‌ریزی می‌شود. دوباره یک روز از time bucket می‌گذرد و دوباره زمان برنامه‌ریزی از آن روز شروع می‌شود. وقتی می‌گوییم frozen window به این معنا است که مثلاً برای هفته‌ی آینده هر چه را گفته‌ایم تغییر نمی‌دهیم. پس هر وقتی من دارم plan می‌کنم، یک هفته را تغییر نمی‌دهم. وقتی زمان یک روز گذشت، این frozen window هم یک روز جلو می‌رود. در این هفته، هر آنچه قبلاً گفته‌ام را دست نمی‌زنم و respect می‌کنم. بنابراین ابتدای plan را که می‌تواند یک روز، دو روز یا یک هفته باشد یا هر عدد دیگری باشد، را تغییر نمی‌دهم. اگر کل time horizon را freeze کنم، من اصلاً Planning انجام نمی‌دهم و همه چیز استاتیک می‌شود. هر چه زمان کم باشد، بسیار داینامیک است و هر چه زمان زیاد باشد به سمت استاتیک می‌رویم، اینجا یک tradeoff است. هر چه داینامیک‌تر باشد، plan حتماً optimal تر است و اجرا سخت می‌شود و برعکس. در مثال گفته شده هر روز، یک هفته از الان به بعد frozen window می‌باشد.

**Holiday Calendar**

برای شرکت‌هایی که بین الملل کار می‌کنند، مثل آمریکای شمالی که در سال فقط ده روز، تعطیلی دارد. این مهم است. شرکت‌هایی که بین‌الملل هستند، باید این موضوع را در نظر بگیرند. من وقتی planning system دارم، باید توجه کنم، که مثلاً ممکن است که حمل و نقل امروز، در یک مکان کار نکند.

**Max lateness**

اگر یک سفارشی backlog باشد، در سیستم تعریف شده است که حداکثر چه قدر می‌توانم تأخیر داشته باشم. بعضی‌ها ممکن است که اصلاً هیچ lateness را نپذیرند. کدام سفارش را ابتدا پاسخ دهیم؟ هر چه max lateness بیش‌تر باشد، فرصت پاسخ‌گویی بیش‌تری دارم.

### Max delivery shipment

ممکن است که مشتری گفته باشد که من حداکثر سه delivery را می‌پذیرم. مثلاً بگویند که من فقط می‌توانم در سه بار، تحویل بگیرم. این موضوع باید در سیستم برنامه‌ریزی اعمال شود.

### Build Ahead limit

فرض کنید که یک سری باکتهای ظرفیت داریم و در هر تایم باکت، ظرفیت صدتایی تولید داریم. یک تقاضا وجود دارد که مثلاً در یک دوره باید تقاضای ۲۵۰ را تحویل بدهم. در نتیجه یک تایم باکت، آن را زودتر آماده می‌کنم و پنجاه عدد باقی مانده را در دو تایم باکت قبلی آماده می‌کنم. به این کار Build Ahead می‌کنیم. مشکلی که پیش می‌آید این است که از فضای Just in time دورتر و دورتر می‌شویم. اگر به اندازه‌ی سه time bucket, build ahead را انجام دادیم، می‌توان از alternate استفاده کرد. در واقع این جا یک limit وجود دارد که نمی‌توان build ahead را به طور طولانی ادامه داد. سوال این است که چه مقدار باید build ahead گذاشت؟ اگر یک جا ظرفیت نداشتیم، می‌توانیم زودتر آماده کنیم. همین‌طور می‌توانیم از جاهای دیگری هم تأمین کنیم.

### Fair-share

در خیلی از صنایع مطرح است. هر جا که assembly باشد، این مفهوم زیاد دیده می‌شود. فرض کنید دو عملیات وجود دارد که این دو یک منبع مشترکی دارند. که inventory target برای این دو عملیات متفاوت است. مثلاً یکی صد و دیگری ۲۰۰ است. در حالی که منبع مشترک به گونه‌ای است که مجموعاً بشود، ۱۵۰ تولید کرد. یک راهکار این است که ۱۰۰ عدد برای اولی و ۵۰ به دیگری داده شود. یعنی یکی را کامل و دیگری را با کمبود، آماده کنیم. ولی در مفهوم fair-share با توجه به نسبت‌ها، برای هر کدام آماده می‌کنیم. در این مثال برای اولی ۵۰ و برای دومی ۱۰۰ عدد آماده می‌کنیم.

### Constrained vs unconstrained planning

مدلی خوبی است که دنیای واقعی را بهتر represent کند. این مفهوم constrained planning وجود دارد. خیلی وقت‌ها، شرکت‌ها می‌گویند که اگر من در دنیای ایده‌آلی بودم، چه قدر می‌توانستم تأمین کنم؟ در اینجا، منابع نامحدود بود. تمام leadtime‌ها را رعایت می‌کنند و ... مثلاً فرض کنید در دانشکده منبع هیئت علمی، هیچ محدودیتی ندارد و می‌توان از تمام دنیا، هیئت علمی آورد. علاوه بر این مشکل، بالاخره من محدودیت زمان و فضا دارم. در آن واحد، دانشکده می‌تواند نهایتاً ۵۷۰ دانشجو را آموزش دهد. این کمک می‌کند که چشم باز شود و فرصت‌هایی که در نتیجه‌ی capacity expansion به وجود می‌آید، دیده شود. دو تا consequence نامطلوب وجود دارد: اول اینکه No Build Ahead شکل می‌گیرد. وقتی ظرفیت زیاد می‌شود، دیگر نیاز نمی‌شود که من وارد time bucket های قبلی شوم. و این خیلی بد است. دوم این که فقط مسیرهای primary route برنامه‌ریزی می‌شود.

این که چه قدر build ahead limit باشد، خودش یک مسئله‌ی opitimization می‌باشد. که پیچیدگی را زیاد می‌کند.

به خاطر این دو consequence نامطلوب این است که از دنیای واقعی دور می‌شویم. کاری که صنعت کرده است (شرکت texas instruement ) این است که ابتدا constrained برنامه‌ریزی می‌کند. وقتی که constrained برنامه‌ریزی کرد، یک سری از سفارش‌ها یا late یا short برنامه‌ریزی می‌شوند. بعد از این مرحله، می‌آیند فقط سفارش‌هایی که short یا late بوده‌اند را undo می‌کنند و از دنیای عملیات بر می‌دارند. در مرحله سوم می‌آیند به صورت unconstrained برنامه‌ریزی می‌کنند. در صنعت به این smart unconstrained planning گفته می‌شود. هر چه منبع هست را به طور حداکثری استفاده می‌کنم و وقتی می‌بینم چاره‌ای ندارم به صورت unconstrained عمل می‌کنم. اینجا می‌بینم که وقتی bottle neck را رفع کنم، می‌توانم چه قدر توسعه در کار دهم.

## Production Planning System

چرا در کشور ما planning هیچ جایگاهی ندارد؟

فرآیند planning این است که ابتدا آخرین وضعیت استاتیک ساختار را می‌بینیم (latest static structure). آخرین وضعیت demand را می‌بینیم. آخرین وضعیت supply side را می‌بینیم. این state of system در واقع وضعیت موجود است. در کشور ما مشکل این است که وضعیت سیستم معلوم نیست. چون ERP وجود ندارد. در ادامه‌ی فرآیند، بالانس بین تقاضا و عرضه بررسی می‌شود. سپس این مراحل با یک فرکانسی تکرار می‌شود. در ایران خودرو و سایپا هم این مشکل وجود دارد، هیچ وقت از وضعیت اطلاعاتی وجود ندارد. مثلاً این که آیا سیستم رباتیک خراب است یا نه؟

مپنا باید وضعیتش مشخص باشد تا بتواند project plan را انجام دهد. در چنین حالتی به جای برنامه‌ریزی به طور دینامیک، به طور استاتیک پیش می‌رود. خیلی از شرکت‌ها این مشکل را دارند که از وضعیت کنونی اطلاعاتی ندارند.

از منظر responsiveness و efficiency می‌خواهیم یک بالانس ایجاد کنیم و در نتیجه همیشه یک سری tradeoff‌هایی وجود دارد و دائم باید tradeoff کنیم تا به هدف business برسیم.

### چالش‌ها در راستای برنامه‌ریزی تولید

#### چالش اول permanent Demand-Supply Balance :

مثلاً وقتی یک قالب شکسته می‌شود، پژو ۲۰۶ دو هفته دیرتر به دست مشتری می‌رسد. سپس مذاکره شکل می‌گیرد و در نهایت گفته می‌شود که از یک تأمین کننده‌ی دیگر تأمین شود. ولی این تأخیر دارد. مثلاً یک هفته این فرآیند طول می‌کشد و بعد از یک هفته، تازه سیستم به سمتی می‌رود که سمت تقاضا با سمت عرضه match شود. اینجا تازه سیستم از دنیای آنچه برنامه‌ریزی شده است خارج می‌شود. اینجا باید سیستم را دوباره replanning کنیم. هر روز و هر ساعت، یک سری chaos‌های پیش‌بینی نشده، رخ می‌دهد. خارج شدن plan از execution ویژگی این سیستم‌ها است.

ERP چیزی نیست مگر این که هر روز، اتفاقاتی که رخ داده است را به صورت manually وارد کنیم. حال این سیستم اگر ساده باشد، یک نرم افزار ساده کافی است ولی در شرکتی مثل مپنا به دلیل این که سیستم خیلی پیچیده است، نیاز است که سیستم های اطلاعات قوی وجود داشته باشد.

اگر یک قلب تپنده نباشد که سازمان را سنکرون نکند (به خاطر تمام رویدادهای پیش‌بینی نشده) حالت سازمان‌های ما رخ می‌دهد که همه همدیگر را متهم می‌کنند. سیستم‌ها باید بتوانند پیوسته خودشان را adjust کنند. این سیستم planning است که به صورت guideline پیش می‌رود. مهم این است که صرفاً matching انجام شود، بلکه می‌بایست به صورت efficient این matching رخ دهد که این کار پیچیده‌ای است. مثل وقتی است که رانندگی در باران و مه می‌کنید. دائماً state را می‌بینید و خودتان را adjust می‌کنید.

حالا که پدیده‌های ناخواسته به طور مکرر شکل می‌گیرد. هنر این نیست که یک adjustment انجام دهیم تا برنامه feasible باشد. feasibility پایین‌ترین سطح برنامه‌ریزی است. ما می‌خواهیم که planing ما optimal باشد. هنر این است که ببینیم برای پنجاه‌تایی که کم داریم، همه‌ی چیزها را سرچ کنیم و تمام راه‌ها را ببینیم و اگر چیزی پیدا نکردم، بگویم نمی‌توانم.

### چالش دوم: Ensure an Optimal D/S Balance

supply demand matching چگونه ماهیت optimization پیدا می‌کند؟ من چندین تأمین‌کننده دارم، سپس چند plant وجود دارد. بعد از آن DC ها وجود دارد و در نهایت به مشتری می‌رسد. این که از کدام ساپلایر گرفته شود، مهم است. مسیری که برای انتقال استفاده می‌شود، در هزینه‌ها موثر است. تکنولوژی plant خودش مهم است. همین که برای هر بخش، گزینه‌های مختلف وجود دارد، combination های مختلفی وجود دارد. آیا من به دنبال efficiency هستم یا responsive ؟ بسته به این‌ها می‌توانم مسیر مختلف را انتخاب کنم.

خیلی وقت‌ها، تقاضا flat نیست. مثلاً سال تحصیلی جدید که شروع می‌شود. ماه مبارک که شروع می‌شود. ممکن است که تقاضا زیاد شود. آیا من باید بیش‌تر از تقاضا تولید کنم؟ یا کمی زیر تقاضا تولید کنم؟ یا کمی بیش‌تر از تقاضا تولید کنم؟ یا حتی flat تولید کنم که تقاضا را به صورت کل پاسخ گو باشم و در زمان‌هایی که بیش از تقاضا تولید می‌کنم، هزینه‌ی موجودی دهم و در زمان‌هایی که تقاضا بیش از تولید است، هزینه‌ی overtime دهم. tradeoff بین این هزینه‌ها، خودش یک optimization problem است و به همین سادگی optimization matching نمی‌توان انجام داد.

فرض کنید resource ای داریم که حداکثر ظرفیت تولید مثلاً صد قطعه در روز است. یک سری محصولات می‌خواهم تولید کنم که بیش از ظرفیت می‌شود. این ظرفیت محدود را به کدام محصول اختصاص دهم؟ این مسئله‌ی تصمیم‌گیری است. ظرفیت یک مقدار ثابت است و حتی ممکن است که اتفاقی بیفتد مثل شکستن متنه که ظرفیت را کم کند. این‌جا باید دوباره adjust کنیم و این یعنی replanning.

Down-Binning Structure: خیلی وقت‌ها یک عملیات تست وجود دارد. ممکن است که پنجاه درصد گرید بالا، ۴۰ درصد گرید پایین و ۱۰ درصد گرید پایین‌تر نتیجه دهد. خیلی وقت‌ها در صنعت Down-binning Structure داریم. برای هر سه گرید گفته شده، تقاضا داریم. اگر تقاضا برای گرید بالا بود، فقط می‌توان از

هر چه دینامیک سیستم بیش‌تر باشد، فرکانس adjust کردن و پیوسته planning کردن کم‌تر می‌شود. هر چه قدر در فضای uncertain هستیم، نیاز به planning بیش‌تر می‌شود. مهم این است که دائماً بتوانیم خودمان را adjust کنیم.



گريد بالا تأمين كرد. ولي اگر براي گريد پايين، تقاضا بود و ما كمبود محصول داشتيم، مي‌توانيم از گريد بالا تأمين كنيم و جواب مشتري را بدهيم. خيلي وقت‌ها وقتي مشتري معطل شود، بار هزينه‌اش بيش‌تر است. اين يك مسئله‌ي *fully fully optimization problem* است. در صنعت نيمه رساناها، ديده مي‌شود كه اگر اين مسئله را optimum كنند، ممكن است كه ميليون دلاري تغييرات ايجاد شود.

در اين مسئله، ممكن است كه وقتي دو تقاضاي يكسان براي گريدها جمع شود، ولي ميزاني كه بايد تأمين كرد ( $A^*$ ) كمتر از جمع  $A^*$  دو حالت قبل شود.

يك مثال ديگر: فرض كنيد كه يك تقاضا وجود دارد كه براي محصول نهايي ۱۰۰ براي ۲۵ روز ديگر واحد تقاضا وجود دارد و زمان از تبديل شدن raw material به محصول نهايي ۲۰ روز طول بكشد. من ۱۲۵ موجودي در حال حاضر دارم. دو گزينه وجود دارد:

گزينه‌ي اول اين است كه همين موجودي كه وجود دارد را به تقاضا بدهيم و اين مصداق *efficiency* است چون از شر موجودي رها مي‌شويم. ولي اگر يك تقاضا دوباره بيايد ممكن است كه نتوانيم پاسخ‌گو باشيم. گزينه‌ي دوم اين است كه پنج روز بعد از تقاضا، شروع به توليد براي پاسخ به تقاضا كنيم. در اين حالت، مصداق *responsiveness* است، چون موجودي را نگه داشته‌ايم كه هر وقت لازم بود به تقاضا پاسخ دهيم. رويكردهاي تركيبي هم وجود دارد كه مثلاً هفتاد در صد را اكنون پاسخ دهم و مابقي را توليد كنم.

يك مسئله‌ي ساده‌ي *inventory netting* يك مسئله‌ي پيچيده‌ي *optimization* است. بايد بلافاصله اثر تغيير را در سيستم ببينيم و بفهميم كه چگونه *Globally* عمل كنيم. اين *optimization* هم مي‌تواند مطلوب نباشد. يك سطح مي‌خواهيم *operational optimization* انجام دهيم. خيلي وقت‌ها نياز است كه *financial optimization* انجام دهيم. در اينجا كار پيچيده مي‌شود و ديگر ذهن مدير و كاغذ و .. هم پاسخ‌گو نيست و نياز به *planning* هاي پيچيده است.

فضا به سمتي است كه به شدت تنوع آن و پيچيدگي آن، زياد مي‌شود. و نياز به *planning* به شدت زيادتر مي‌شود. حتي در *strategic planning* هم همين طور است. اگر بخواهيم *strategic planning* هم درست باشد، بايد مرتب در طول سال تكرر شود و خودمان را *adjust* كنيم و در واقع خودمان را با ديناميك صنعت *adjust* كنيم. در سيستم‌هاي مديريت يك سيستم وجود دارد به نام *Order Management System*.

در مقام رويكردها و الگوريتم‌ها، رويكردهاي مختلفی وجود دارد. يكي از رويكردها، رويکرد سنتي *linear programming* است. حالت ديگر *heuristic* ها مي‌باشد كه پر از *if then condition* مي‌باشد. مدل‌هاي ديگر هستند كه از نوع *search* هستند. موارد نظير *genetic algorithm*، *fuzzy logic* و... در اين جا مطرح است. در صنايع مختلف وقتي صحبت از صنعت مي‌شود، *Linear Programming* هميشه حرف اول را مي‌زند. ويژگي كه *LP* را منحصر کرده است، اين است كه *LP* گارانتی می‌کند كه برای شما *Global*

حقيقت اين است كه در دنياي *planning* خيلي *objective* هاي مختلف ظاهر مي‌شود. خيلي از منافع با هم در تضاد هستند. بعضي از چرخ‌دنده‌ها بر عكس هم عمل مي‌كنند. مثلاً نمي‌توان هزينه‌ها را كم كرد و *responsive* بود. در *project management* هم همين طور است. هزينه، زمان و كيفيت سه متغيري است كه هميشه در *conflict* مي‌باشند.

Optimization انجام دهد. موارد دیگر وابسته به Initial Condition می‌باشند که با تغییر شرایط اولیه، جواب کلی تغییر می‌کند.

خیلی از وقت‌ها، فضا اصلی linear نمی‌باشد. یک disadvantage بزرگ برنامه‌ریزی خطی دارد این است که خیلی وقت‌ها مثل ایرلاین‌ها، اصلاً linear نمی‌باشند. مثلاً نمی‌توان گفت  $\frac{3}{2}$  هواپیما. اما، همه‌ی دردرس‌هایش می‌ارزد. چون Global optimal را ضمانت می‌کند. معمولاً بعد از حل LP، یک فرآیند Post-Processing انجام می‌دهند تا شرایط را تطبیق دهند.

مواردی نظیر MIP (Mixed Integer Probleming)، فقط به صورت لوکال، بهینه‌سازی انجام می‌دهد ولی به صورت Global انجام نمی‌شود. علتی این که صنعت Linear Programing انجام می‌دهد، همین موضوع است. هیچ روش دیگر Global Optimization را ضمانت نمی‌کند. به خاطر سادگی و فهم، نسل‌های قدیمی، ممکن است که ترجیح دهند که از Heuristic ها استفاده کنند و یک disadvantage دیگر LP این است که مثل یک Black Box می‌باشد و خیلی قابل لمس نمی‌باشد که چه اتفاقی می‌افتد.

روش‌های Heuristic به شدت، به Initial Condition وابسته می‌باشند و این برای صنعت خیلی نامطلوب است. برای صنعت Reliability بسیار مهم است. می‌خواهند اگر یک جواب را صد بار هم run کردند، به یک نتیجه برسند.

شرکت سامسونگ، از برنامه‌ریزی LP که برایش انجام شده بود، بسیار ناراضی بود. چون هر Data set که بش داده می‌شد، یک جواب کاملاً متفاوت می‌داد. مشکل این بوده است که از DataBase اشان و نحوه‌ای که داده‌ها گرفته می‌شد، sequence گرفتن اطلاعات جابه‌جا می‌شد. در این‌جا جواب Optimal ثابت است ولی cornerها تغییر می‌کند. به لحاظ ریاضی کار درستی مدل شده بود ولی به لحاظ دنیای عملیات، جابه‌جایی زیادی رخ می‌داد. در نهایت، کاری کردند که جواب به جواب corner قبلی نزدیک باشد.

Linear Programming برای aggregate planning مناسب است. اگر یک operation ساده وجود داشته باشد و مشخص شده باشد که در time bucket های مختلف، چه قدر باید تولید شود و اگر تقاضاهای مختلف وجود داشته باشد، دقیقاً مشخص نمی‌شود که در یک time bucket کدام محصول برای کدام تقاضا تولید می‌شود. (این اشکال دیگر LP است) در Post Processing می‌توان این کار را انجام داد. به هزینه‌ی تعداد متغیرها و زیاد کردن Constraint ها می‌توان این مسئله را حل کرد که بدانیم محصولی که تولید می‌شود برای کدام تقاضا است ولی مسئله این قدر بزرگ می‌شود که نمی‌توان آن را حل کرد.

عیب Heuristic ها این است که بسیار suboptimal می‌باشد و نمی‌توان به صورت financial optimization انجام داد.

فقط با LP می‌توان Financial Optimization انجام داد. در Financial Planning ، OrderByOrder نمی‌توان، برنامه‌ریزی کرد. تا Plan انجام نداده باشم، اصلاً معلوم نیست که Cost چه قدر است. من وقتی می‌دانم هزینه‌ی تمام شده چگونه است که بدانم چگونه plan انجام شده است و priority سفارش‌ها را اصلاً نمی‌توان مشخص کرد. برای این‌که بتوانم به روش‌های دیگر برنامه‌ریزی کنم، باید OBO ها را داشته باشم، و برای داشتن این‌ها باید هزینه‌ها را داشته باشم و تا plan انجام نشده باشد، نمی‌توان هزینه‌ها را داشت. در برنامه‌ریزی نمی‌توان order by order عمل کرد. LP در آن واحد، گلوبالی هزینه‌ها را هم می‌بیند.

در صنعت، سائز مسائل صنعت بزرگ است. یک سیستم زنجیره تأمین اگر بخواهد واقعاً به درستی represent کنیم، سائز بزرگی دارد و برای سائز بزرگ فقط LP است که می‌تواند جواب دهد. وقتی مسئله بزرگ می‌شود، روش‌های دیگر خیلی سریع گیر می‌کنند و همان نقطه شروع را می‌گویند بهینه است. علی‌رغم این‌که خیلی از چیزها طبیعت linear ندارد و integer می‌باشند، به خاطر ویژگی‌هایش از LP استفاده می‌شود.

ویژگی دیگر LP که کمک می‌کند در دنیای مدیریت از آن استفاده کنیم:

z1: Max demand set (shortness | >) ==> Dimesion = Unit

z2: Min Lateness ==> T

z3: Max Rev ==> \$

ذهن ما نمی‌تواند، این‌ها را همزمان با هم sync کند. (در واقع هر کدام از مسائل بهینه‌سازی، بعدهای مختلف دارند). در دنیای آکادمی برای هر از این‌ها یک وزنی مشخص می‌کنند ولی در دنیای عمل این طور نیست. در دنیای عمل باید از مدیران بپرسیم و این خیلی متفاوت است. اگر از یک مدیر که تازه چک امضاء کرده است، می‌گوید درآمد خیلی مهم است. از مدیری که چک امضاء نکرده است بپرسیم می‌گوید که lateness مهم است چون به برند آسیب می‌رساند.

$$Z=w1*z1+w2*z2+w3*z3$$

در صنعت، می‌آیند به طور hierarchical عمل می‌کنند. می‌پرسند که اگر از کل این‌ها بخواهید یکی را انتخاب کنید کدام را انتخاب می‌کنید؟ مثلاً می‌گوید: lateness. بعد از او دوباره پرسیده می‌شود که حالا که lateness را انتخاب کردیم، بعد از آن چه چیزی مهم است؟

این موضوع را به راحتی در LP می‌توان دید:

$$\max Z1 \quad Z1^*$$

$$\text{st: } Z1 > Z2 > Z3$$

$$\max Z2$$

Searched based  
برنامه‌ریزی‌ها، برای کارهایی  
است که خیلی به صورت  
detaied است و مشکل این  
است که این روش‌ها به شدت  
در suboptimal ها، گیر  
می‌افتند.

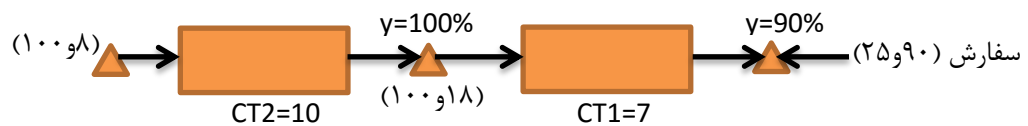
در دادن وزن نسبی ذهن  
انسان‌ها inconsistant عمل  
می‌کند و در heirarchical به  
صورت consistant عمل  
می‌کند.

st: c1

$$Z2 = Z1^*$$

به صورت مرحله‌ای عمل می‌کنیم ولی هر مرحله، به معنای بدتر شدن مرحله‌ی قبل نیست. ما هر کدام را بگیریم برای ما علی السویه است. در هر مرحله بدون این که قبلی degrade شود یک constraint اضافه می‌کنیم. این کاری است که در صنعت انجام می‌شود و به شدت هم رایج است و عملاً خیلی مفید است. وقتی objectiveهای مختلف داریم، نمی‌توانیم از وزن‌های نسبی استفاده کنیم. روش‌های Heuristic، هم order by order planning رخ می‌دهد. همه را یک مرحله بالاتر می‌بریم و زنجیره را جاری می‌کنیم. مثلاً شرکتی که می‌گوید max profit به صورت یک objective می‌باشد. این صد در صد در تعارض است با max inventory targets. داشتن موجودی یعنی هزینه و با objective که می‌گوید سود را ماکزیمم کن، در تعارض است. اگر اول بخواهم سود را ماکزیمم کنیم و سپس inventory targetها را، هیچ وقت، دومی رخ نمی‌دهد. رویکردهای مختلفی این جا وجود دارد که وارد جزئیات آن‌ها نمی‌شویم.

وقتی یک network پیچیده باشد، از LP استفاده می‌کنیم. منظور از پیچیده این است که گزینه‌های زیادی برای برنامه‌ریزی وجود دارد. زنجیره تأمین خیلی عمیق است و این‌ها همگی دال بر پیچیدگی است.



در این مسئله مثلاً اگر capacity هم shared شود، ولی مسئله پیچیده می‌شود. هر چه alternate و ظرفیت باشد، پیچیدگی زیاد می‌شود. در این سیستم‌ها خیلی وقت‌ها، انواع مختلف الگوریتم‌ها استفاده می‌شود.

یک مثال از Hierarchical planning :

فرض کنید که چهار objective وجود دارد. دیده می‌شود که در لایه‌های پایین حل این مسئله، space همین طور cut می‌شود. این قدر feasible space در این جور مسائل بزرگ است که می‌توان، همین طور narrow down کرد و به پاسخ بهینه رسید. راجع به response buffer قبلاً صحبت شده بود. بالا دست push و پایین دست pull بود.

علاوه بر مدل‌های گفته شده، یک مدل دیگر ETO یا Engineer to Order می‌باشد که برای تجهیزات پیچیده است. برای تجهیزات سنگین، ماشین‌های CNC، تا سفارش داده می‌شود، تازه تأمین کننده می‌رود سفارش‌هایش را می‌دهد.

سوال این است که برای ما در شرایط مختلف، کدام گزینه بهترین است؟ دو بعد وجود دارد: بعد اول: استاندارد بودن محصول یا *vairability* آن است. بعد دوم: طول *operation* ما چه قدر طولانی است یا کوتاه؟ وقتی *variability* کم است و محصول استاندارد است، و *lead time* زیاد است، خوب است که MTS عمل کنیم. وقتی *leadtime* کم است ولی محصول به شدت *customized* است، مفهومی وجود ندارد که MTS عمل کنیم. در اینجا MTO خوب است. در دو حالت دیگر این کوآدرانت، بستگی به شرایط می توان MTS یا MTO باشد. مثلاً اگر یک محصول جدید *launch* می شود، خوب است که MTS باشیم.

## Traditional Production Planning - MRP

در تمامی کتابها، این قسمت قسمت اصلی آن است. این خیلی ساده است .

MRP= Material Requiring Planning توسط IBM معرفی شد و سریع گسترش یافت. اساس این Bill of Material و یک سری *fiexed cycle time* می باشد.

فرض می شود که تقاضا *deterministic* و معلوم است. ظرفیت نامحدود می باشد. *production lead time* معلوم و ثابت است.

ورودی ها:

Bill Of Material

Cycle Time

Demand Input

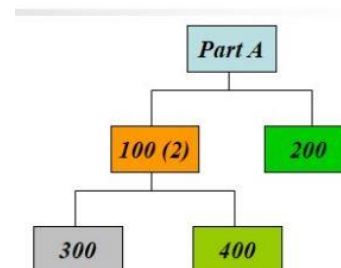
Inventory Statues

Schedule Receipts

برنامه ای است که سمت تقاضا دریافت می کنیم.

خروجی، برنامه ی مد نظر است.

روش این است: ابتدا *netting* انجام می دهیم.



در نمودار یک BOM در بالا، عدد (۲) مقدار است. inventory netting یک مسئله optimization است. مرحله‌ی بعد از netting باید، lot sizing انجام داد.

نسل دوم MRP، یک سری منابع را وارد می‌کند. ما جدای از MRP، می‌آییم منابع را به صورت aggregate چک می‌کنیم. دیده می‌شود که باز هم نسل ۲، با دنیای واقعی تفاوت دارد. این گونه نیست که بگویم کارخانه این قدر ظرفیت تولید دارد. همیشه fluctuation ها وجود دارد. این که overall ظرفیت سالیانه را ببینیم، نمی‌تواند برای تصمیم‌گیری مناسب باشد چون با دنیای واقعی و پیچیدگی‌هایش متفاوت است. تا حالا وقتی گفته می‌شود، با تسامح می‌گفتیم که یک نوع planning داریم.

#### لایه‌های مختلف برنامه‌ریزی برای تولید

planning لایه‌های مختلف دارد:

Corporate planning چندین division را در بر می‌گیرد و چندین کارخانه را در نظر می‌گیرد. جزئیات مدل، خیلی aggregate می‌باشد. هدف از این برنامه‌ریزی، این است که ببینیم چه قدر منابع محدوداند یا زیادی داریم. هر چه قدر ببینیم، ظرفیت چه قدر نیاز است می‌باید با subcontractor ها قرارداد ببندم. می‌خواهم ببینم برای forecast ام، roughly چه قدر باید factory اضافه کنم.

لایه‌ی Master Planning خیلی خیلی غالب است و بحث‌های اصلی ما در این کانتکست بوده است. زمان آن خیلی وقت‌ها ۶ تا ۸ ماه می‌تواند باشد. تفاوت با قبلی این است که صرفاً برای یک division می‌باشد. مثلاً فقط برای memory division برنامه‌ریزی می‌شود. یک division می‌تواند چندین کارخانه را در نظر بگیرد. هدف این است که ببینم که تقاضا چگونه تأمین می‌شود و در مواد اولیه، چه چیزهایی نیاز دارم. هر کدام load balancing را به طور گلوبال انجام می‌دهد.

هم ممکن است که جنبه‌ی مالی و هم عملیاتی وجود داشته باشد.

Factory Planning: برای یکی از کارخانه‌ها، جزئیات عملیات را باز می‌کنم و هی کلیک می‌کنیم و زوم این می‌کنم و جزئیات را به طور ماکرو می‌بینم. برای یک کارخانه زمان می‌تواند در حد چند هفته باشد. برای چند division ممکن است مصرف باشد. این برنامه‌ریزی بیش‌تر جنبه‌ی عملیاتی است و کم‌تر توجه وجود دارد که چه قدر این کارخانه برای من پول در می‌آورد.

این که بده بستان بین این‌ها در صنعت چگونه است، به شدت پیچیده است.

close loop، به شدت مهم است. مثلاً وقتی در factory می‌خواهیم برنامه‌ریزی کنیم، گفته می‌شود که من در لایه‌ی بالا کلیات را مشخص کردم، تو برو جزئیات را مشخص کن. در سیستم‌های غربی این close loop بودن خیلی مهم است ولی در شرکت‌های شرقی بیشتر ماهیت top down وجود دارد.

## Master Planning

یک سری محدودیت‌ها و یک سری business objectives نظیر این‌که می‌خواهیم responsive باشیم یا نباشیم، به آن داده می‌شود. خروجی این است که ببینیم تقاضا چگونه تأمین می‌شود؟ سفارش چگونه تأمین می‌شود؟ ATP یعنی Available To Promise می‌باشد. مثلاً اگر ۱۰۰ عدد پیش‌بینی شود و ما بتوانیم، ۸۰ عدد را پاسخ‌گو باشیم، به این مورد ATP گفته می‌شود. اصلاً forecast را داریم که بعدها plan کنیم و order وارد سیستم شود. وقتی وارد Advanced Planning system شویم، این مورد خیلی شفاف‌تر می‌شود. مثلاً در سایت آمازون هر سفارشی که شکل می‌گیرد، قبلاً برای آن plan انجام شده است و می‌تواند که پاسخ‌گوی این سفارش باشد و این به معنای الزاماً physical on hand نمی‌باشد و این در سیستم order management است.

در سیستم planning آن نگاه end to end، در ایران وجود ندارد. علاوه بر این‌که state سیستم را نداریم، این مشکل وجود دارد.

## Distribution

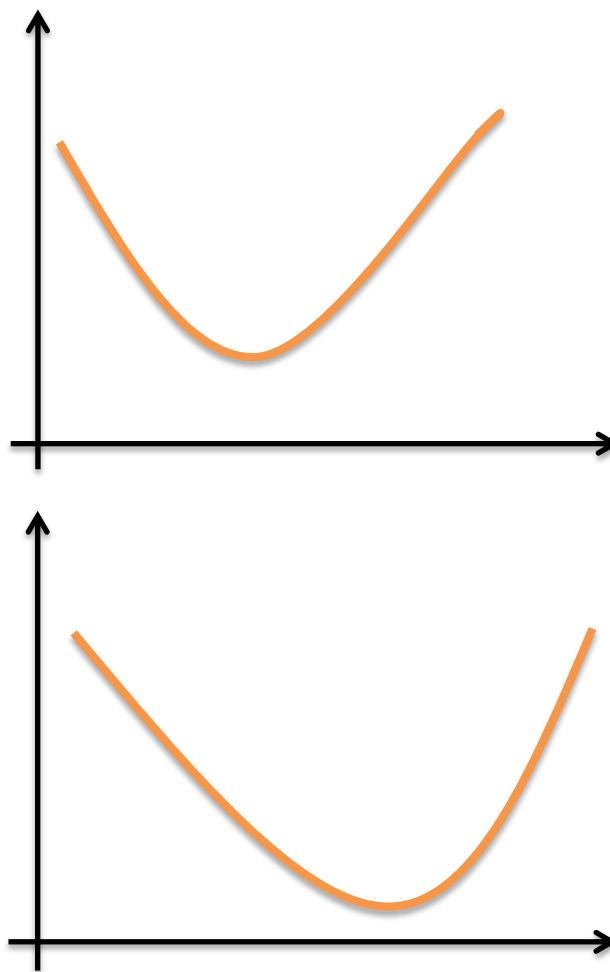
یک محصول ساخته شده است و تولید شکل گرفته است. مدیریت موجودی و procurement هم شکل گرفته است. الان می‌خواهیم، محصول را به بازار برسانیم. سوال این است که خودمان این کار را انجام دهیم یا دیگری؟

به لحاظ تئوری و عملی اگر بخواهیم مقایسه کنیم:

وقتی یک تولید کننده وجود دارد، کالا را به عرضه‌کننده می‌دهد تا به بازار عرضه کند. این اگر بخواهد کارش را خوب انجام دهد و core competency تولید را داشته باشد و میزان production را داشته باشد، زمانی خوب عمل می‌کند که از نظر economic of scope و scale در جای خوبی ایستاده باشد. این شرکتی است که خوب عمل می‌کند و شرط لازم برای فضای رقابتی را می‌تواند داشته باشد. یک نمودار برای تولید کننده وجود دارد. اگر عین این نمودار را برای توزیع کننده، بکشیم، نمودار دیگری وجود دارد. همان‌طور که در نمودارها دیده می‌شود، وقتی تولید کننده بهینه است، توزیع کننده بهینه نیست. لذا اگر تولید کننده خودش بخواهد توزیع کند، حتماً صرفه‌ی مقیاس در توزیع ندارد.

نمودار برای تولید کننده و توزیع کننده در ادامه آمده است. (محور عمودی: هزینه، محور افقی: تعداد کالا)

خوب است که ابتدا supply chain را به صورت end to end برنامه‌ریزی کنیم و بعدها هر کدام از کارخانه‌ها و جاهای مختلف را به صورت دقیق‌تر plan کنیم. این‌که تکنولوژی خاصی برای برنامه‌ریزی یک بخش از زنجیره تأمین بدون در نظر گرفتن end to end کار مناسبی نیست.



زمانی توزیع‌کننده خوب عمل می‌کند، زمانی است که تولید‌کننده به صورت بهینه تولید می‌کند. از نظر تئوری کسی که تولید می‌کند، هیچ دلیلی وجود ندارد که توزیع هم بکند.

وقتی کالا، توزیع را خودش انجام می‌دهد، زمانی به صرفه است که لجستیکی که وجود دارد، بتواند خیلی از محصولات را انتقال بدهد. در این حالت، توجیه اقتصادی دارد. در بسیاری از مواقع این دو مورد خیلی نمی‌تواند یکی باشد. اما در دنیایی که الآن هستیم، شرکت توزیع قوی نداریم، چون به شدت به تکنولوژی وابسته است و سیستم برنامه‌ریزی قوی می‌خواهد. بنابراین چون توزیع‌کننده‌های حرفه‌ای در ایران نداریم، شرکت‌های داخل می‌گویند با این که می‌دانیم efficient نیستیم، توزیع را خودمان انجام می‌دهیم و این تفاوت دنیای عمل و تئوری است.

فضای distribution با فضای production کاملاً متفاوت است. چون در production می‌توان به راحتی افراد را دید ولی در distribution ممکن است که افراد هزار بهانه‌ی واهی مثل ترافیک بیاورند. distribution



می‌تواند یک برند را آسیب بزند یا آن که آن را به ارج برساند. مثلاً وقتی کالا بهترین NDP انجام دهد، ولی توزیع بد باشد، تلاش بیپرده است. پیش‌رفت تکنولوژی، به بهبود این مسئله کمک می‌کند.

service design و product desing کردن دارد به سمت commodity می‌رود. شرکت‌هایی هستند که برای شما تولید و بسته‌بندی را انجام می‌دهند. مهم detect کردن سیگنال از مارکت است و distribution و پایین دست آن retailing این فضا را مهیا می‌کند.

به شرکت‌های تولیدکننده همیشه توصیه می‌شود که به نوعی دسترسی به بازار را حفظ کنند. اگر چه هزینه‌هایش بالا است، ممکن است که بتواند سیگنال بازار را دریافت کند. برای شرکت‌های بزرگ، گفته می‌شود که حداقل ۱۵ تا ۲۵ درصد، خودش از طریق کانال مستقیم ارائه دهد. امروزه فضای آنلاین هم به عنوان یک کانال جدی مطرح است. در فرآیند اینکه اپل، اپل شد، توزیع‌کننده‌ها نقش بسیار مهمی داشتند.

## distribution roles

توزیع‌کننده سه نقش ایفا می‌کند:

- اینکه در یک سبد، چه ترکیبی از محصولات بگذاریم؟
- کالا را در زمان و محل مناسب ارائه دهیم.
- Market assessment و این یعنی عملاً با مشتری تعامل داشتن و البته core عملیات که توزیع است را انجام دهد.

خیلی وقت‌ها، فکر می‌کنیم که شرکت‌های توزیع می‌بایست فقط کالا تحویل دهند. این، بهانه است و فقط می‌تواند از این طریق پول روزنامه را در بیاورد. توزیع‌کننده حرفه‌ای، دو وظیفه‌ی دیگر را تمرکز می‌کند. مثلاً توزیع‌کننده است که به تاجر می‌گوید که چه نوع عسل را ببار. توزیع‌کننده، می‌داند که بازار چه چیزی را می‌خواهد. این اصلاً کار جذابی نیست که یک کسب و کاری فقط توزیع انجام دهد و پنج درصد، مارجین داشته باشد. این فضا، مانند فضای گوگل است. گوگل به افرادش حقوق‌های بالا می‌دهد تا موتور جستجو تقویت کند و در واقع به نوعی از بازار، insight داشته باشد. خیلی وقت‌ها، شرکت‌ها مثلاً می‌گویند که من این کالا را برای رایگان حمل می‌کنم. اینجا به data دسترسی پیدا می‌کنند و این خیلی ارزشمند است. در صنعت دارو ممکن است که کل تحویل دارو، مجانی باشد ولی با یک database مرکزی باشد. این داده‌ای که به دست می‌آید این قدر ارزشمند است و می‌تواند تعداد زیادی کسب‌وکار را ارائه دهد. این جا می‌داند که چه داروهایی مثلاً خوب کار می‌کنند و کدام کار نمی‌کند؟ کدام زود خراب می‌شود؟ و... کسی که دسترسی به بازار دارد، جایگاه ویژه‌ای دارد.

ممکن است که یک فرد، بگوید من می‌خواهم کالایی بیاورم و برای این که چه کالایی بیاورد، ۵۰۰ میلیون حق مشاوره به کسی که به insight از بازار دارد، پرداخت کند.

هر چه زمان می‌گذرد، صاحبان برند از نظر production نقش کم رنگ تری دارند پیدا می‌کنند و به بازار access دارند. کسی که retailing و تعامل با مشتری دارد، جایگاه بسیار بسیار ویژه‌ای دارد چون می‌تواند از بازار، insight بسیار مهمی به دست آورد.

### نقش اول: Combining right portfolio of products

من باید، بدانم که چه محصولات و چه برندها با چه کیفیت و قیمتی را می‌خواهم ارائه بدهم. Market research هم به طور جدی در کشور ما مطرح نمی‌باشد. این که کالا از کجا وارد شود، هزینه‌ی حمل و نقل و موجودی چه قدر است، یک جنبه‌ی دیگر است. این که چه کسانی می‌توانند مشتری من باشد و بازار چه قدر فضایش برای من جذاب است؟ همه‌ی این‌ها می‌شود، پورتفولیو را چیدن.

### نقش دوم: Delivery of products in a right time to a right place

همان کار pure distribution است. من می‌دانم چه کالایی را انتخاب کردم و باید یک سری Customer satisfaction انجام شود و بتوانم عملاً چیزهایی که لازم است انجام شود را به خوبی انجام دهم. واکنش‌های مشتری را باید رصد کرد که اگر مثلاً کالا دیر ارائه شود، promotion خوب ارائه نشود و... چه اتفاقاتی رخ می‌دهد؟ مثلاً مشتری سوئیچ می‌کند.

### نقش سوم: Market assessment

بدانم، سمت مشتری چه چیزی می‌خواهد. بدانم، انتخاب کالا غلط بوده یا نبوده است. به نوعی این نقش به دو نقش قبلی، فیدبک می‌دهد. چون market assessment امروزه خیلی مهم است، حتماً مهم است که امروزه هر شرکت تولیدکننده یک کانال مستقیم داشته باشد، اگر چه از نظر اقتصادی به صرفه نیست.

### سطح‌های توزیع:

#### سطح اول: Network Desing

ماهیت کلان دارد. من کجاها باید به کجاها وصل باشم. کجاها باید انبار و DC بگذارم. یک جاهایی هست که باید کالا را پخش کنم. من برای پاسخ به این تقاضا، چه مسیرهایی را باید طراحی کنیم. چرا به این سطح استراتژیک گفته می‌شود؟ چون ماهیت کلان دارد. این جا کلی، کالا وجود دارد و بنابراین این جا باید، توزیع قوی داشته باشم. مثلاً از اصفهان به تهران می‌خواهم ببرم. چه جاهایی، node بگذارم و انبار داشته باشم؟ نمی‌توان هر روز و هر هفته، این طراحی را تغییر داد. در نهایت، دو ساله بتواند تغییر کند و برای همین استراتژیک است. باید ببینیم که source و sink ما کجا می‌باشد. باید دید که مسیرهای هر کدام، چگونه متصور است.

باید دید که چرخه عمر چگونه است؟ مثلاً اگر قرار است پنیر از آمل بیاید تهران، کدام مسیر را انتخاب کنم که محصول فاسد نشود. این‌ها ملاحظات است که در فضای طراحی شبکه می‌تواند مؤثر باشد.

باید بدانیم که الگوی تقاضا چگونه است؟ fluctuation چگونه است؟

این که من چه Nodeهایی را نیاز دارم. در واقع همان طراحی شبکه است.

مثلاً ذوب آهن اصفهان، در شهرهای مختلف بازار دارد. سوال این است که به تمام این شهرها، مستقیم بفرستد و ممکن است که هزینه بالا رود. یا این که یک انبار بین این شهرها بگذارم، و کالا را از آن به آن‌ها بفرستم. اینجا ممکن است که هزینه بسیار کم‌تر شود.

وقتی گفته می‌شود، طراحی نتورک یعنی این که چه گره‌هایی لازم است و نرخ توزیع کالا چگونه باشد؟ مثلاً سائز انبار چه قدر باشد؟ برای هر لینک، چه قدر material flow نیاز دارم. وقتی حجم یا این ده درصدها را بدانم، یعنی این که چند نتورک می‌خواهم.

در مقام طراحی نتورک، سوال این است که چه قدر دنبال responsiveness هستم و چه قدر دنبال efficiency هستم؟ در طراحی نتورک، انواع گزینه‌ها با توجه به پاسخ این سوال مطرح است. برای این که ببینیم چه عواملی مؤثر هستند، چه قدر تنوع کالا وجود دارد؟ چه قدر availability می‌تواند بالا باشد. یعنی این که کالایی بد بود و رفت از جایی خرید، به راحتی می‌تواند برگرداند؟ مثلاً می‌گوید که من اصلاً چنین پرینتر نمی‌خواستم. حالا فروشنده می‌گوید که من اصلاً معذوریتی ندارم و دست من نیست، در این حالت، خریدار دچار سردرگمی می‌شود. یک نکته‌ی دیگر، visibility است. آیا من هر لحظه می‌دانم، کالا در کجا قرار دارد.

برای response time نگاه کردن، هر چه بخواهیم response time بالا باشد (بار منفی دارد، یعنی دیر پاسخ دهیم)، باید تا می‌تواند نزدیک مشتری باشد. responsive بالا بودن، به معنای هزینه‌ی بالا داشتن است. از منظر response time، هر چه node ها زیاد شود، عملاً response time کم می‌شود (number of facilities).

از منظر حمل و نقل، یک نقطه‌ی بهینه وجود دارد. یعنی با افزایش تعداد node هزینه تا یک جایی کم می‌شود و بعد از آن هزینه زیاد می‌شود. خوب است که من به مشتری نزدیک باشم ولی تا یک حد مناسب است. مثلاً ممکن است که یک سری وانت داشته باشیم که فقط یک محصول را به صورت inefficient حمل می‌کند.

تعداد facility را اگر در نظر بگیریم، می‌توان نمودار response time و هزینه را بر اساس آن ببینیم و بر اساس آن (البته نه نمودار، این نمودار صرفاً هزینه‌ها را نشان می‌دهد)، می‌توان راجع به responsiveness و efficiency صحبت کرد.

شش مدل برای طراحی معماری شبکه توزیع وجود دارد. (در درس، به طور عمیق بررسی نمی‌شود):

نوعی اول: سیگنال از مشتری به خرده فروش می‌رسد و بعد این سیگنال به تولید کننده می‌رسد و تولید کننده به صورت مستقیم به مشتریان می‌دهد. از لحاظ موجودی خیلی مناسب است ولی از لحاظ حمل و نقل، هزینه‌ها به شدت بالا می‌رود.

تلاقی دو نمودار response time و total cost معنایی ندارد.

نوع دوم: In-transit Merge Network: سیگنال‌ها به یک خرده‌فروش مرکزی می‌رود (مانند حالت قبل است) ولی حمل و نقل به صورت مستقیم نیست. بلکه یک انبار مرکزی وجود دارد.

نوع سوم: یک گرهی مرکزی وجود دارد، که هم سیگنال را می‌گیرد و هم کالا را ارائه می‌دهد MTO برای تولید کننده‌ها نمی‌باشد.

نوع چهارم

نوع پنجم، سفارش‌ها به یک‌جا می‌آید، و عملاً این‌ها pickup می‌کنند و همه‌ی این‌ها به یک مرکزی می‌آیند ولی pickup sites را باید انجام دهد.

### سطح دوم: Planning & Schedueling

یعنی از مرحله‌ی قبل می‌دانیم که کجاها node داریم و هر کدام چه قدر ظرفیت داریم. الآن می‌خواهیم وارد planning شویم، یعنی بر اساس forecast و سفارش‌ها باید از هر کجا به کجا، چگونه و چه تعدادی بفرستیم. Scheduling، برای کوتاه مدت و روزانه است. در مقام planning به صورت هفتگی و ماهیانه مشخص می‌کنیم و بر اساس چه سیاستی، چگونه replenishment انجام دهیم. در schedueling به زمان حال نزدیک می‌شویم. باید بدانیم که دقیقاً چه shipment برای فردا باشد و تمام exception‌ها را handle کنیم. اگر کمبود تقاضا در انبار جیرفت داریم، چه کار کنیم؟ از انبار شمالی بفرستیم؟ چگونه آیت‌ها را از لحاظ چیدمان داشته باشیم. آن‌چه در برنامه‌ریزی تولید گفته شد، در توزیع هم می‌تواند استفاده شود.

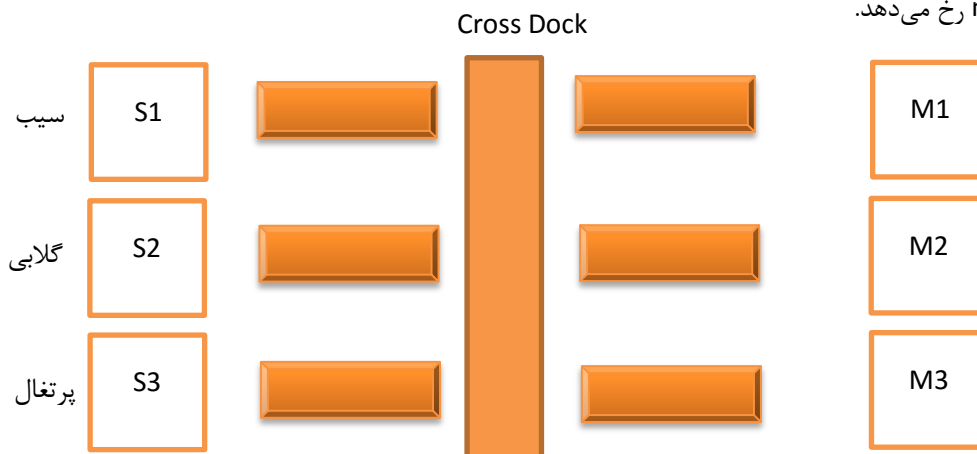
در داخل شهر، حتی در سطح تهران، یک سیستم حمل و نقل مناسبی وجود دارد.

### سطح سوم: Excecution

اجرا، در حوزه‌ی توزیع، دنیای منحصر به فرد خود را دارد. خیلی با دنیای تولید متفاوت است. در planning مشخص است که چه کالایی به چه کامیونی در چه زمانی داده شود. حال در عمل اگر این‌ها به خوبی شکل نگیرد، ممکن است که به یک کالا آسیب وارد شود. وقتی داریم از دنیای operations management صحبت می‌کنیم، فکر می‌کنیم صرفاً روی کاغذ است. این‌که بفهمیم نیروی انسانی من چگونه عمل می‌کند، مهم است. این که بگوییم سیاست‌ها را ما گذاشتیم و بقیه را راننده باید انجام دهد، اشتباه است. مثلاً وقتی بستنی می‌رسد، طرف ممکن است بگوید که من بستنی میوه‌ای می‌خواستم و تقاضاها ممکن است که تا دقیقه‌ی نود تغییر کند. این‌ها همگی از آن دنیای excecution است. مثلاً قبلاً شیشه‌های شیشه‌ای برای شیر وجود داشت و جمع کردن آن‌ها خودش دارای عملیات بود. خیلی وقت‌ها همین‌که بخواهیم یک آیت‌ها را تراک بگذاریم و قرار است پنج جا برود و ویزیت کند، خودش مسئله‌ی بهینه‌سازی است. این‌که کدام آیت‌ها را دم دست بگذارم، کدام را نگذارم، مسئله‌ی بهینه‌سازی است.

سیستم‌های planning  
هوشمند، همه‌ی این‌ها را لحاظ  
می‌کند.

مفهوم Cross Dock: مبدع این مفهوم وال مارت است. تعداد تأمین‌کننده وجود دارد و بازارها در سمت دیگر می‌باشد. ترکیبی از میوه‌ها مثل پرتغال، سیب و گلابی قرار از این سیستم توزیع می‌شود. بنابراین باید ترکیبی از همه‌ی این‌ها فرستاده شود. کاری که وال مارت انجام داد، یک سری سکوهایی را مشخص کرد تا کامیون‌ها بارشان در انبار خالی کنند. سپس یک سری کامیون دیگر می‌آیند، این محصولات را به صورت ترکیبی بار می‌زنند. در واقع به جای این‌که کالا بماند، در یک مکان در اردر یک تا دو ساعت معطلی فرآیند mix & match رخ می‌دهد.



خیلی وقت‌ها اگر خوب plan نکرده باشیم، در دنیای execution به مشکلات زیادی می‌خوریم. مثلاً بسته‌بندی خیلی مهم است. مثلاً آی‌کیا، خداوندگار این است که هر محصول را چگونه بسته‌بندی کنند که هر نفر بتواند آن را بلند کند و بتواند داخل ماشین بگذارد. بعضی وقت‌ها اصلاً جای پارک نیست. در دنیای تولید، این مشکلات وجود ندارد ولی دنیای execution دنیای distribution بسیار پیچیده‌تر است. بسیار شرکت‌هایی هستند که در دنیای manufacturing توانمند هستند و فکر می‌کنند که می‌توانند در دنیای distribution هم موفق باشند یا برعکس. ممکن است که ما یک مرحله را خوب انجام دهیم، ولی کارهای پایین‌دست و بالادست خودمان را نتوانیم به خوبی انجام دهیم.

## Advanced Planning Systems

به شدت کمک می‌کند که بدانیم چگونه شرکت‌ها end-to-end را plan می‌کنند. Planning یک تیغ دو دم است. مثل یک ماشین سرعت بالا است که اگر دست یک آدم بی‌احتیاط باشد، مشکل آفرین می‌شود. اگر در APS بلد نباشم، اصولی را در مدیریت رعایت نکنم، مشکل‌ساز خواهد شد. در دنیای management فکر می‌کنیم که اگر یک سری سیستم‌ها را داشته باشیم، همه چیز خوب کار می‌کند ولی این‌ها شرط لازم است ولی کافی نیست. مثلاً اگر دیتابیس تویوتا را کپی کنیم، ماشین ما تویوتا نخواهد شد. این یک فرهنگ نیاز دارد و چیزی نیست که با کپی کردن، مشکلات حل شود. فقط نباید از بعد تکنولوژی نگاه کرد و توانمندی ما در مدیریت مهم است.

در واقع در cross dock، ما mix و match انجام می‌دهیم. ترکیب محصولات را از کامیون‌هایی که یک سری بار را آورده‌اند در کامیون‌های دیگر می‌گذاریم.

یکی از چالش‌هایی که در سیستم planning وجود دارد، data می‌باشد. کیفیت data مهم است و این در حوزه‌ی ERP می‌باشد. dataهای گذشته را باید بخوانیم و بفهمیم که پارامترها چگونه است و این به شدت به data که از ERP می‌آید، وابسته است. خیلی وقت‌ها اگر بخواهیم سیستم به خوبی Planning شود و انعطاف‌پذیر باشد، هزینه‌ی Data بسیار زیاد می‌شود.

چالش دیگر این سیستم‌ها، اگر APS، Upgrade شود، خودش یک پروژه‌ی مفصل است و اگر Data، Upgrade شود، APS هم باید Upgrade شود و این خودش از چالش‌های جدی است. در یک پروژه از زمینس، سه سال متوالی طول کشید تا این داده‌ها integrate شوند. خیلی وقت‌ها ریشه‌ی مشکلات در data management می‌باشد. خیلی وقت‌ها شرکت‌ها نیاز به flexibility در داده‌هایشان دارند و این جا به خاطر هزینه‌های داده، ممکن است ترجیح بدهیم که کل برنامه با یک data انجام شود.

جنبه‌ی دیگر چالش این سیستم‌ها، implementation این سیستم‌ها می‌باشد که خیلی سخت است. معمولاً tradeoff بین optimization و heuristic مطرح است. به این معنا که وقتی می‌گوییم optimize یعنی یک سیستمی که کاملاً بهینه است ولی ممکن است سیستمی را بخواهیم که جعبه سیاه نباشد و بدانیم هر لحظه چه اتفاقی می‌خواهد رخ دهد. معمولاً مدیران می‌خواهند بدانند که وضعیت چگونه است. اما با گذشت زمان، شرکت‌ها یاد گرفته‌اند که خوب است کارها را فازبندی کنیم و از تجربیات هر فاز برای فاز دیگر استفاده کنیم. مثل compound interest است که از هر سود، دوباره سود بگیریم.

Small Vs. Large Corporate: فضای شرکت‌های بزرگ و کوچک با هم متفاوت است و باید این تفاوت را دانست. در شرکت‌های بزرگ، معمولاً یک پروسه وجود دارد و برای جلوگیری از خسته می‌شوند. شرکت‌های کوچک چابک‌تر می‌باشند و یک نفر را که قانع کنید، کافی است. از منظر implementation به همان راحتی که می‌توان مدیرعامل یک شرکت کوچک را می‌توان قانع کرد، نظرش عوض می‌شود. در شرکت کوچک، نظر ممکن است که مرتب عوض شود ولی در شرکت‌های بزرگ با این کار بروکراتیک است، ولی وقتی تصمیمی گرفته شد، ادامه‌ی مسیر، مرتب تغییر نمی‌کند.

New systems vs. Old approaches: داستان change management است. تغییر تحول را نباید دست کرم گرفت. اگر بیست سال، یک فرد یک کار را انجام می‌داده است و از فردا محصول شما بخواهد، تغییر ایجاد کند، فرد ممکن است این کار را انجام ندهد. تغییر تحول در خود سازمان هم مهم است.

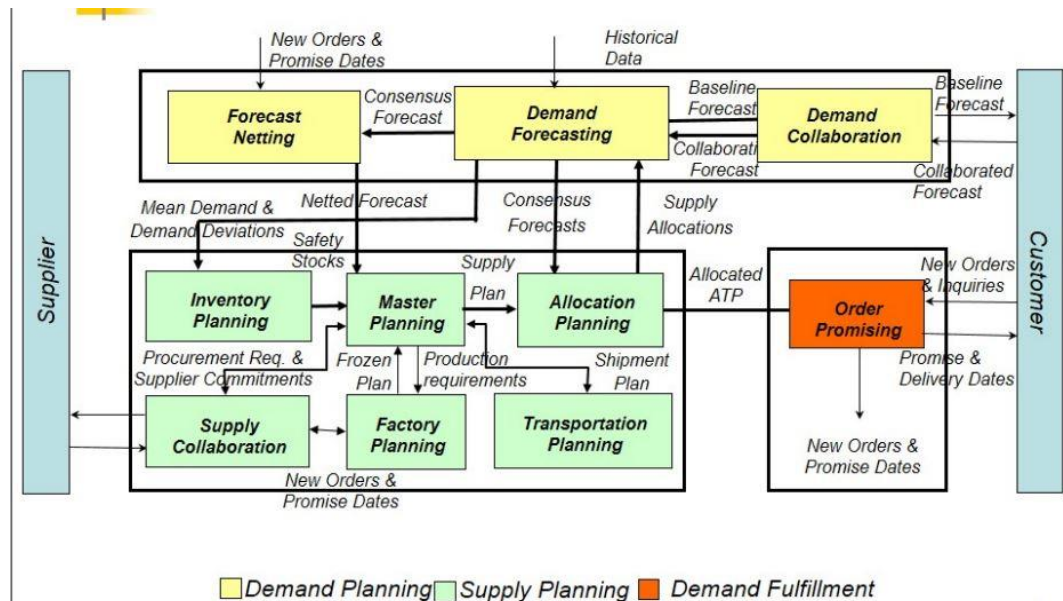
در این سیستم‌ها، انعطاف‌پذیری ممکن است که مهلک باشد و هر چیزی یک هزینه دارد. و هزینه یعنی اطلاعات بیش‌تری را نگه داشتن و در واقع پیچیدگی بیش‌تر. از نظر خود پیچیدگی (complexity)، ما فکر می‌کنیم که اولین نفرات هستیم و می‌خواهیم چرخ را از ابتدا اختراع کنیم ولی باید ببینیم که دیگران چه کار کرده‌اند. این در دیار ما، بسیار مشکل‌آفرین بوده است. این اشتباه است که فکر کنیم، چون زمینس این کار را کرده است، مپنا هم می‌تواند انجام دهد.

legacy system ها، در واقع همان ERP داخل یک شرکت است.

چرخ را نباید از ابتدا اختراع کنیم ولی باید localization و بومی‌سازی را انجام دهیم ولی این خیلی انرژی‌بر است.

شرکتی که تاکنون APS نداشته‌است، همه‌ی گزینه‌ها را می‌خواهد داشته باشد، انواع گزارش‌ها را می‌خواهد ولی این کار اشتباه است. چون یک مدت این‌ها را نداشته‌است، همه چیز را می‌خواهد، در نتیجه می‌بایست به مرور زمان انواع گزینه‌ها را وارد کند.

End-to-end زنجیره تأمین بدین شرح است که سمت پایین دست سمت مشتری است و سمت بالا دست، تأمین‌کننده وجود دارد. این جا سه نوع ماژول وجود دارد.



یک ماژول، حوزه‌ی مدیریت تقاضا (Demand Management) است. از جنس demand planning است. ماژول دیگر، supply planning است. خیلی وقت‌ها فکر می‌شود که مسئله این‌جا تمام است ولی این‌گونه نیست. ماژول سوم، Demand fulfillment مطرح است. باید بتوانم در دنیای واقعی، demand fulfillment به صورت real time انجام دهم. ممکن است که برنامه‌ریزی به صورت هفتگی شکل گرفته باشد ولی در دنیای واقعی به صورت real time شکل می‌گیرد.

در ماژول Demand Planning:

Demand Collaboration: ممکن است که مشتری خودش از محصول، ۲۰۰۰ آیتم درخواست کند. ممکن است که در پاسخ گفته شود که ما حتی بیشتر برای آماده کرده‌ایم، این جا تعهدی شکل نمی‌گیرد و صرفاً visibility وجود دارد. از برآیند خیلی عوامل، به این نتیجه می‌رسیم که forecast مثلاً یک مقداری خواهد بود. به این حالت Baseline Forecast گفته می‌شود.

Demand Forecast: برای یک محصول، مثلاً این T-shirt و این سایز، forecast من، این مقدار است.

Forecast Netting: نهایتاً نظرات تیم بازاریابی، فروش، مشتریان به یک نظر کلی می‌رسد. مثلاً از یک محصول، این قدر پیش‌بینی وجود دارد و به این consensus forecast گفته می‌شود. مثلاً از صبح تا حالا، ۵ تا تقاضا آمده است و باید تحویل بدهم. Forecast اولیه ام، ۱۰۰۰ تا بوده است، در این جا، ۵ عدد firm order است و در سیستم planning ۹۹۵ باقی‌مانده، جزء forecast باقی می‌ماند.

#### کیس BMW

کاری که در کیس BMW انجام شد، این بود که video streaming شکل گرفته بود. اولین بار ولوو، خانواده را موقع مونتاژ کردن سر خط تولید دعوت می‌کرد و فرزند خانواده نسبت به ساخته شدن ماشین، احساس داشت. اینجا داستان تجربه به صورت جدی مطرح است. امروزه ماشین‌های نسبتاً جدید، گفته می‌شود که تفاوت آنچنانی ندارند. تفاوت ملموس وجود ندارد. بنابراین این که ماشین فی نفسه بخواهد از لحاظ تکنولوژی تفاوت و تمایز ایجاد کند، تقریباً به آخر خط رسیده است.

آنچه در این کیس مطرح می‌شود، این است که این experience می‌تواند متفاوت باشد. آوردن این مفهوم از دنیای بازاریابی به دنیای supply side ظرافت می‌خواهد. حالا که این فضا مطرح می‌شود، چه اتفاقی رخ می‌دهد. یا مثلاً فرض کنید از روی موبایل و آی‌پد، بتوان این نوع سفارش محصول را انجام داد. مثلاً در ایران دامینو پیتزا، این قابلیت را آورد که خانواده‌ها که می‌خواستند پیتزا سفارش بدهند (مخصوصاً برای بچه‌ها خیلی جالب بود) بتوانند سایز و محتویات را مشخص کنند. این فضا دارد کم‌کم روتین می‌شود و ارتباط دنیای order management و دنیای planning گسترده‌تر می‌شود. مثلاً شکلات‌های N&N در سفارش می‌توانید اسم خودتان یا دوستان را روی شکلات بگذارید. عملاً در وبسایت خط تولید این را برای شما مشخص می‌کند. خط تولید با order management system می‌تواند با بخش تقاضا با هم link شود.

سطح customization که بخواهد اتوماتیک هم باشد، باز هم خیلی پیچیده نیست از لحاظ رباتیکی. آنچه که پیچیده است ارتباط این موضوع با supply chain می‌باشد. در واقع این قسمت از رباتیک سخت‌تر است. رباتیک را با تکنولوژی حتی در جهان سوم می‌توان به آن دست یافت ولی فرایند ارتباط این موضوع‌ها با همدیگر، به این راحتی‌ها نیست.

موجودی در شرکت هم کم شده بود. خود این یک رنوسانس است. بالاخره بعضی از رنگ‌ها و آپشن‌ها را مشتریان نمی‌خواهند. وقتی تولید کننده customization را انجام می‌دهد، ممکن است که پروژه با شکست مواجه شود. چون سود distribution را داریم از بین می‌بریم. عده‌ای این وسط دارند، مارجینی را می‌بینند و حذف می‌شوند و ممکن است که شیطننت خودشان را بکنند. در عین حال که دنیای supply chain با دنیای demand side بسیار متفاوت است مخصوصاً از دیدگاه مدل کسب‌وکار. هر چه قدر customization بالا رود، ارتباط با بخش supply side ظرافت بیش‌تر دارد. مایکل دل، مبدع این داستان بود و گفت که من اتفاقاً می‌خواهم در فضای رقابتی وارد شوم و customization را به طور جدی اعمال کنم. بنابراین این سوئیچ کردن به این سطح customization برای کانال پیامد دارد. بنابراین اگر برای تمام خودورها این کار را انجام دهیم، ممکن است که کل channel آسیب ببیند.

اگر این تکنولوژی را در تمامی خودروها توسعه بدهیم، شدنی است؟

به نظر می‌رسد که بازار از این موضوع استقبال کرد. از سمت demand side و از سمت supply side چالش‌هایی وجود دارد. ظاهراً مدل‌های x با هر شماره برای یک سگمنت تارگت شده‌اند. مثلاً x3 برای پز دادن



برای نوجوانان است. در حالی که سری‌های دیگر X وجود دارد، که اصلاً این موضوع مهم نیست. مثلاً یکی از مدل‌هایش برای مدیران ارشد و CEO ها است. خیلی از این مدل‌ها را ممکن است که اصلاً افراد نخرند و corporate ها بخرند. بنابراین target market متفاوت است. اگر این‌گونه باشد، کانال مشکلی نخواهد داشت. یک سری سگمت خاص را به کانال می‌دهیم و یک سری سگمت خاص که عاشق show off می‌باشد، را به صورت custom ارائه می‌دهیم.

کتاب The more the less توسط شوارتز نوشته شده است و می‌گوید که آپشن بیش‌تر ممکن است که مشتری را ناراضی کند. یعنی وقتی شما، آپشن‌های زیادی دارید، ممکن است رضایت شما کم شود. آدم‌ها دو نوع‌اند یا maximizer یا satisficer. این‌ها نه بد است نه خوب است، این‌ها personality type انسان‌ها است. از نظر انتخاب کاراکتر بعضی‌ها maximizer است و بین گزینه‌ها دوست دارند که بهترین را انتخاب کنند. در عمل معمولاً خوشحال نیستند و معمولاً ارضا نمی‌شوند. چون یک کالا را هنوز استفاده نکرده، می‌بینند که مدل بهتر آمده است. این آدم‌ها در نهایت خوشحال نمی‌باشند. در تقابل با این سگمت، افرادی هستند که satisficer هستند و هر چیزی که enough باشد را می‌پسندند. این‌ها از افزایش تعدد و customization لذت می‌برند. بنابراین این‌ها که بتوانیم در دنیای supply chain، customization را بالا ببریم، برای این سگمت مفید است.

در سیستم و فرهنگ غربی، این variety یک چیز جا افتاده است و اگر محدود شوند، ناراحت‌اند. در فرهنگ ژاپنی، معتقدند که باید میزان variety را محدود تعریف کنند. سه نوع ماشین SUV ارائه می‌دهند و از هر کدام دو سه نوع آپشن ارائه می‌دهند. می‌گویند این هشت نه مدل، کافی است و این برای آن‌ها good enough است.

ما خیلی وقت‌ها در فعالیت‌هایمان good enough را خوب است به جای maximization مد نظر داشته باشیم. در حوزه نرم‌افزار، ژاپنی‌ها دنبال zero defect هستند و برعکس آن‌ها آمریکایی‌ها دنبال good enough می‌باشند و میکروسافت با مشکلاتی که داشته است، موفق بوده است. در دوران لانچ هر محصول، دانشکده یا سازمان، باید به این ویژگی بسیار توجه کرد. مثلاً برای لانچ دانشکده ممکن است که یک سری استاد بگیریم و به مرور زمان، این مشکل را رفع کنیم. دینامیک هر محصول و خدمت جدیدی است.

سمت supply side شاید بتوانیم کارهای زیادی بکنیم. ولی وقتی بخواهیم با demand side سنکرون شویم، باید حد دینامیک بودن که از سمت تقاضا بر supply demand می‌گذارد، رعایت کنیم. این‌ها مثل VAS می‌مانند. دیده می‌شود که چرا خیلی از feature ها را اضافه نمی‌کنند؟ چون برای مخابرات و تولیدکننده‌ها هیچ سودی ندارد. مثلاً عکس پاناروما را شرکت‌های تولیدکننده موبایل نمی‌گذاشتند. بعد از این که به مرور زمان، نرم‌افزارهای مختلف برای این کار ارائه داده شد، این جزء native feature موبایل‌ها شد.

برای BMW X3 هم باید، به همین شکل نگاه کرد. چشم به هم بگذاریم ممکن است که دهی دیگر، همه‌ی شرکت‌های تولیدکننده این را به کار خودشان اضافه کرده‌اند. دیر یا زود این می‌تواند یک native feature برای تمام commodity ها باشد.

افرادی که BMW می‌خرند، معمولاً ماشین نمی‌خرند، بلکه برند می‌خرند. قیمت برای این محصول، اصلاً ویژگی نیست. ۲۵ درصد از profit margin را BMW به R&D می‌دهد. ساختار سازمان بی‌ام‌و، مستقل است. شرکت‌هایی مثل مرسدس بنز، تحت هولدینگ است که ممکن است فشارهای مختلفی به آن‌ها وارد شود. طراحی محصول در BMW خیلی مخصوص است و محصولش از این بابت متفاوت است. بنابراین بخشی که

می‌خواهد عرضه کند، سگمندی است که به قیمت کم‌تر حساس می‌باشند. استراتژی در این کیس، saving های زیادی می‌آورد. موجودی در سیستم کم می‌شود و با وجود Initial investment بالا، saving زیادی دارد. دنیای Business Model، دنیایی است که باید در MBA به طور مناسب تمرین کرد. مدل می‌تواند توجیه سرمایه‌گذاری اولیه را داشته باشد.

ماژولاریتی نوعی postponement می‌باشد. این گیربیکس می‌تواند در یک ماشین عادی استفاده شود و در یک ماشین دیگر. هنر این است که بتوانیم چگونه برنامه‌ریزی کنیم که چه موقع چه چیزی را ارائه دهیم. در واقع platform based عمل می‌کنیم. من دو سه نوع object دارم. بعضاً یک کد یا object می‌نویسم و آن را در جاهای مختلف استفاده می‌کنم. بنابراین تناظر یک‌به‌یک بین محصول خروجی و کامپوننت‌ها می‌تواند متفاوت باشد. در نرم‌افزارها هم ماژولاریتی داریم و کدهای تقویم یا security یا report دادن با copy و paste کردن اضافه می‌کنند. فقط چند خط کد اضافه می‌کنند که این‌ها به هم وصل شوند.

#### ماژول Supply Planning:

Inventory planning، ورودی‌های زیادی دارد و متشکل از شبکه‌های مختلف است. خروجی این، inventory target در زنجیره تأمین است و این یک ورودی به master planning است و master planning لایه‌ای است که برای هر Business Unit، planning را به طول گلوبالی انجام می‌دهیم. ورودی دیگر master planning از سمت supply Collaboration می‌آید. در واقع بر اساس تعهد supplier، یک base line برای forecast به دست می‌آید.

Factory Planning وجود که ورودی‌اش، Frozen window و WIP می‌باشد. و این یک ورودی برای master planning است.

ورودی دیگر برای master planning، سیستم حمل و نقل می‌باشد. این لزوماً توزیع نیست، مثلاً من باید بین واحدهای خودم، حمل و نقل را مدیریت کنم.

Allocation Planning: آنچه که forecast است و ما plan می‌کنیم. مثلاً برای خرداد ۱۰۰ forecast وجود داشته است و امروز، سوم خرداد ۲۰ سفارش آمده است، netted forecast ۸۰ می‌شود، دوباره وقتی Plan می‌شود، به جای ۸۰ ممکن است، ممکن است با توجه به محدودیت‌ها ۷۰ plan کنم. آیا همه‌ی این‌ها را در اختیار مشتریان قرار می‌دهم؟ دلیلی وجود ندارد که این کار را انجام دهم. اگر یک finished goods وجود داشته باشد، انواع orderها هستند که اولویت دارند و یک سری هم forecast وجود دارد. فرض کنید که این forecastها ۱۰۰ بوده است و ۲۰ سفارش آمده است و الآن netted forecast ۸۰ می‌باشد. حالا من در plan به خاطر ظرفیت‌ها، ۶۰ عدد را plan می‌کنم. این ۶۰ عدد Available to Promise می‌شود. آیا اگر در سیستم مثل دیجی‌کالا، این ۶۰ عدد سفارش داده شود، همه دیده می‌شود؟ ممکن است که این سیستم یک allocation را انجام داده باشد. گفته می‌شود که این Available to Promise می‌باشد اما من فقط ۴۰ تا را به

مشتریان که از قبل نمی‌شناسم اختصاص می‌دهم و می‌خواهم ۱۵ عدد از آن را به big customers می‌خواهم اختصاص بدهم. من اگر ۶۰ تا در اختیار دیگر مشتریان بگذارم، ممکن است که اگر سفارشی از مشتریان بزرگ بیاید، پاسخ‌گو نباشم و آن‌ها ناراضی باشند. ۵ عدد دیگر از محصولات را می‌خواهم به خارج از تهران اختصاص دهم. در این حالت، آنچه order to promise است، AATP می‌باشد و با توجه به AATP به مشتری قول می‌دهم. آیا این بر اساس موجودی است که دارم؟ خیر، بر اساس planning می‌باشد و من می‌دانم که چه مقدار می‌توانم تحویل دهم. وقتی یک شرکت قول می‌دهد که مقداری کالا تحویل دهد، feasibility را از قبل بررسی کرده است. این موضوع Demand fulfillment است. در سیستم‌هایی که سایت‌های مختلف دارند، این سه ماژول، مطرح است. Demand Fulfilment، real time می‌باشد و round clock من باید، سفارش دریافت کنم.

همیشه ما real time که سفارش می‌گیریم، بر اساس چیزی سفارش می‌گیریم که حتماً می‌توانیم آن را تحقق می‌بخشیم. این به معنای این نیست که در سیستم موجودی داریم نیست. این به معنا این است که می‌توان در طی یک هفته از الآن می‌توانم مثلاً صد آیتم، از یک محصول را داشته باشم. چون plan کردم، می‌دانم که یک هفته‌ی دیگر در دست خواهم داشت و می‌توانم اگر سفارشی بیاید، پاسخ‌گو باشم. نود درصد موارد برای سفارش‌ها نباید مشکل داشته باشیم و اگر به محدودیت بخوریم، از ظرفیتی که برای فورکست است، دچار مشکل می‌شویم.

دینامیک صنعت پیوسته تغییر می‌کند. ما در کشوری هستیم که پروژه‌های صنعتی زیاد داریم. ما باید به ERP وصل باشیم و بدانیم که State پروژه در هر لحظه چگونه است. اگر State پروژه را نداشته باشیم، نمی‌توانیم دینامیک عمل کنیم.

## Demand Planning

خیلی وقت‌ها Horizon شش ماه تا یک سال دارد. ممکن است که ما در یک صنعت باشیم که دینامیک کم باشد ولی در Master plan، نیاز به forecast داریم و این ورای زمان محدود است.

Planning bucket از هفتگی تا یک ماه تغییر می‌کند.

قابلیت‌هایی که این سیستم باید داشته باشد:

پدیده‌forecasting، یک پدیده‌ی چند بعدی است و باید این را handle کنیم. سه بعد Customer، زمان و مشتری وجود دارد که هر کدام از این‌ها سلسله مراتب می‌باشد. بنابراین وقتی می‌خواهیم Forecast کنیم، سه Hierarchy باید در هم ضرب شود. از لحاظ بستر IT، این یک بستر پیچیده است.

بنابراین همیشه real time که سفارش می‌گیریم، بر اساس چیزی می‌گیریم که بتواند به سفارش تحقق بخشد. این به این معنا است که من یک هفته از الآن می‌توانم مقدار زیادی از این داشته باشم. در همه‌ی سیستم‌ها این ساختار قابل قبول است.

تکنولوژی‌های نظامی نظیر وصل کردن RFID به لباس‌ها، در جاهای CIVIL وارد شده است تا هر لحظه از وضعیت سیستم با خبر باشیم.

در Forecasting در هر یک از این Hierarchy ها، می توان از پایین به بالا و از بالا به پایین عمل کرد. خیلی وقت ها گفته می شود که مثلاً در این فصل، Target به این شکل است. شما به ما بگو که امسال چه کاره اید. همیشه نتایج این دو روش پایین به بالا یا بالا به پایین با هم ناسازگار است.

این چنین Forecast که می خواهد شکل بگیرد، آدم های مختلف involve هستند. Salesforce، یک نقش ایجاد می کند. باید consensus forecasting انجام داد. خود مشتری یک سیگنال می دهد. تیم بازاریابی، یک ورودی می دهد. خط تولید، تیم فاینانس همگی یک نتیجه می دهد و اعداد هر کدام از این ها با هم متفاوت است. اگر از Sales Force سوال پرسیده شود، همیشه Overestimate می کند، چون دنبال کمسیونش است. از تولید پرسیده شود، همیشه underestimate می کند تا بعد بگوید که من در عمل بیشتر تولید کردم.

ما برای یک مشتری استراتژیک، چه قدر پیش بینی می بینم؟ این عدد را قرار نیست که همه ببینند. بنابراین این کار security خاص خودش را می طلبد.

Forecasting باید انواع Forecasting technique ها را در بگیرد. از گذشته، بعضی وقت های میانگین متحرک وجود داشته است، سری های زمانی وجود داشته است که نمی توانسته است دینامیک را به طور دقیق پیشنهاد دهد. رویکردهای مختلف وجود دارد و خوب است که یک سیستم forecasting تمامی این موارد را در نظر بگیرد.

باید بتوان قابلیت Product substitute داشته باشیم. خیلی وقت ها کارهای Forecasting بر اساس Historical Data می باشد. ممکن است که یک محصول، کاملاً جدید باشد ولی به خاطر بعضی از ویژگی هایش مثلاً صفحه نمایش، شبیه محصولات دیگر باشد. Forecast دیگر محصولات را دارم و می توانم با ضربی از آن پیش بینی برای این محصول پیش بینی کنم. وقتی می توانم این کار را بکنم که قدرت Product Substitution وجود داشته باشد.

Rich set of allocation policies: وقتی دارم Top Down عمل می کنم بتوانم هر لحظه Allocate کنم و به پایین بروم. مثلاً کل فروش ده هزار تا برای کشور، مقداری اش برای شرق و مقداری اش برای غرب است.

Collaboration Capabilities: وقتی می دانیم پیش بینی هر کدام چه قدر است، بتوانم با موارد دیگر collaboration انجام دهم.

Seasonality , life cycle and promotion: مثلاً یک روز به خاطر یک رویداد قرار است تخفیف گذاشته شود، پس این یعنی Forecast بیش تر. همین طور ممکن است که فصل هایی باشد که تقاضا زیاد شود.

چالش هایی که سیستم Forecasting دارد:

خانم سیمون ری، معتقد است کسانی اخلاق مدارتر هستند که قدرت Imagination بیش تری داشته باشند. کسی می تواند این قاعده را رعایت کند که خودش را در جای دیگران قرار دهد.

اولاً اساس Forecast ، Customer Requested Date باید باشد. اگر هزار نفر می‌خواهند بیایند در دانشکده ثبت نام کنند ولی ظرفیت وجود ندارد، عیبی ندارد. باید Forecast را بر این اساس انجام داد. CRD محدودیت است.

#### Measuring Forecasting accuracy:

در واقع خطای پیش بینی به نوعی مبنایی برای Forecast می‌باشد. هر چه در سطوح بالاتر بروم، چون Aggregation رخ می‌دهد، خطا کمتر می‌شود. من دقیق‌تر می‌توانم بگویم که امسال چند جفت کفش می‌توانم بفروشم ولی به راحتی نمی‌توانم بگویم که از کفش قهوه‌ای این مارک، چه تعداد می‌فروشم.

#### Arriving at consensus forecast:

تمام شرکت های خوب، جلساتی دارند تا به Consensus برسند. این جلسات زمان‌بر است و هر کس می‌خواهد Position خودش را حفظ کند. در نهایت به یک تصمیم‌گیری نهایی می‌رسند.

#### Lengths of historical data:

من تا چه قدر زمان قبل را بروم بررسی کنم؟ هر چه قدر این بازه زمانی بیش‌تر شود، یعنی وزن Recent data کم‌تر می‌شود.

#### Direct relation with pricing policies:

ذهنیت قدیمی بود که می‌گفت ببینیم Forecast کنیم و ببینیم که چه قدر تقاضا است. امروزه گفته می‌شود که پیچ تقاضا در دست خودمان است. قیمت را پایین ببر، تقاضا زیاد می‌شود و تقاضا یک متغیر مستقل نیست. امروزه Price را به عنوان یک متغیر مستقل می‌بینند. امروزه جنبه‌ی Price خیلی مهم است. هرچه که در گذشته انجام شده است، می‌تواند اصلاً مصداق نباشد.

بنابراین امروزه خوب است که demand elasticity را forecast کنیم. این را که به دست آورم، قیمت را هر چه ست کنم، در واقع با تقاضا دارم بازی می‌کنم.

### Inventory Planning

دوباره variable time bucket است و Time bucket ها بین ۶ ماه تا یک سال است. می‌بایست ABC Classification انجام داد. این سیستم باید بتواند به Target های مد نظر را بش برسد. هزینه را باید به خوبی maintain کند. باید عملاً تقاضا و Variability را respect کنم.

از ویژگی‌های مهم، scenario management می‌باشد. سناریوی خوشبینانه و بدبینانه را باید در نظر بگیرم. باید خود قابلیت اصلی را جواب بدهم. باید بتوانم Financial analysis انجام دهم. باید بتوان budget

optimization انجام داد. باید بتوان override انجام بدهم. در اینجا سیستم ممکن است که یک عدد بدهد ولی من باید با تصمیم‌گیری شهودی راجع به این عدد قضاوت کنم.

چالش‌ها:

### Capacity Constraint

در برنامه ریزی، مثلاً گفته می‌شود که من نهایتاً می‌توانم ۱۰۰۰ تا تولید کنم، پس inventory target ها چگونه است.

### Master Planning

لایه‌ای است که end to end زنجیره تأمین را به هم وصل می‌کند و جایگاه بسیار ویژه‌ای دارد.

در حال حاضر، Inventory Planning و master planning را به صورت جدا می‌بینیم. چرا این‌ها را یکی نکنیم؟ این‌جا مسئله‌ی گلوبالیزیشن مطرح است که اگر این‌ها با هم باشند، می‌توان به صورت گلوبالی عمل کرد و چه لزومی دارد که این‌ها را جدا کرد؟ اصلی‌ترین دلیل، محدودیت تکنولوژی است. این قدر مسئله planning پیچیده است که اگر بخواهیم این کار را بکنیم، اصلاً تکنولوژی قابلیت پاسخ دادن به این مورد را ندارد.

### Master Planning

شش ماه تا یک سال است.

ویژگی‌ها:

### Hard Constraint Vs. Soft Constraint

#### Hard Constraint

مواردی نظیر lead time و yield است.

Soft Constraint این است که مثلاً تعداد تحویل من از ۳ بار کم‌تر باشد.

در حوزه‌ی Frozen window می‌شود گفت که حالت Hard Constraint این است که بگوییم هیچ گونه تغییر ندهیم حتی اگر Feasible نباشد ولی Soft می‌توان بگوییم که تا حدودی امکان تغییرات وجود دارد.

چالش‌ها:

### Optimization vs. Heuristic

گفته بودیم که LP جایگاه ویژه‌ای دارد ولی خیلی از چیزها هستند که گسسته می‌باشند. این چیزی است که پازل اصلی می‌باشد.

از بین تمام ماژول‌هایی که وجود دارد، Demand Planning یک جایگاه ویژه‌ای دارد. دلیلش این است که ورودی بقیه‌ی ماژول‌هاست و اگر چشمه خراب باشد تا آخر رودخانه آب گل آلود است. اگر بد عمل کنم، همه‌جا مشکل دارم. بنابراین هر چه در Demand Planning effort بگذاریم، ارزشمند است.

دو محور پیچیدگی و optimized plan وجود دارد. هر چه قدر پیچیدگی heuristic را بالا ببریم، می‌تواند تا یک سطحی از Optimization بالا رود. اما با پیچیدگی زیادتر، f then condition ها زیاد می‌شود و نمی‌توان گفت که چرا این جواب به دست آمده است.

اما LP کلاً به صورت black box می‌باشد و رابطه‌ای با complexity ندارد.

Split of large models: وقتی مدل‌های بزرگ وجود دارد، معمولاً مدل را Split می‌کنند.

Priority assignment for demands: گفته بودیم که اولویت Forecast همیشه از order کم‌تر است. استثنای این حرف وقتی است که یک سری سفارش داریم برای زمان آینده ولی یک سری Forecast وجود دارد که به زمان حال نزدیک است. سوال این است که سفارشی که برای چهار ماه دیگر است را اولویت قرار دهیم یا Forecast که برای امروز است. وقتی که forecast نزدیک باشد، می‌توانم ATP داشته باشم و از آن سفارش دریافت کنم.

WIP netting policies:

اگر استراتژی responsive بخواهم داشته باشم، باید چیزی که در WIP است را نگاه دارم و از ابتدا plan کنم. (یعنی در نظر نگیرم)

Run time

Incremental planning

اگر ما planning را دوباره انجام دهیم، ممکن است که از نظر بیزینسی نتایجی بدهد که کاملاً متفاوت باشد. (مانند مثال سامسونگ)

Factory Planning

جنسش خیلی شبیه Master Plan است ولی برای یک کارخانه می‌باشد. جزئیات بیش‌تری نسبت به master planning دارد.

یادمان باشد وقتی factory planning انجام شد، به این معنا نیست که می‌توانیم آن را انجام دهیم و هنوز scheduling باقی مانده است. اگر رویدادی بوده است و .... باید به صورت روزانه بررسی کنیم.

باید بتواند جزئیات زیادی را Handle کند و زمانش از چند روز تا چند هفته است.

چالش عمده‌ای که این‌جا وجود دارد، integration بین master planning و factory planning است. در اولی جزئیات کمتری داریم ولی در دومی جزئیات بیش‌تری دارد. دانشگاه می‌خواهد برای دانشکده‌ها planning کند. دانشگاه به صورت کلان برای هر دانشکده نگاه می‌کند اما دانشکده هم می‌خواهد planning انجام دهد، جزئیات را باید مدل کند. این که چه تعداد دانشجو accumulated شده‌اند. این جا یک بده‌بستان شکل می‌گیرد.

## Demand Fulfillment

به صورت روزانه تا هفتگی است.

ویژگی‌ها:

باید بتواند عین مسئله‌ای که در demand side وجود دارد را این‌جا هم داشته باشیم. باید hierarchy ها را بفهمیم و ATP ها را apply کنیم.

rich set of allocation: آیا همه‌ی محصولات را ارائه دهد یا نه بعضی از محصولات را مثلاً برای دانشگاه جدا بگذارد.

Due date quat mechanism: خیلی وقت‌ها در demand fulfillment، سفارش را در نظر می‌گیریم ولی زمان لازم است تا یک quate به order تبدیل شود.

### کیس 711

شرکت سال ۷۰ میلادی تأسیس شده است و بسیار قدیمی است. در سال ۹۷، ۱۶ میلیارد دلار درآمده داشته است. یک retailer بسیار قوی بوده است. retailing بسیار اهمیت دارد. به خاطر این‌که به بازار دسترسی دارد. در گذشته این چنین نبوده است ولی امروزه کسانی که دسترسی به بازار دارند، می‌توانند بازار را جابه‌جا کنند. ژاپن به مساحت کم و جمعیت بالا معروف است. در آمریکا اگر نگاه کنیم، جایی است که زمین گسترده است و این قدر محدودیت در فضا نیست.

New open store strategy این گونه است که فروشگاه به گونه‌ای سِت نمی‌شوند که با هم رقابت کنند. در مارکت با دیگران رقابت می‌کنند و خود فروشگاه‌ها با هم رقابت نمی‌کنند. اشکالش چیست که overlap نداشته باشد از لحاظ operation چه ویژگی دارد؟ خوب است که این فروشگاه‌ها با هم رقابت نکنند یا نکنند؟ کدام رویکرد بهتر است؟ مثلاً استارباکس سر هر کوچه یک مغازه وجود دارد و فروشگاه‌ها با هم رقابت می‌کنند. اگر رقابت نباشد performance هر کس فقط با خودش مقایسه می‌شود که چه قدر از دیگران (رقبا) مشتری می‌گیرد. اگر selection اولیه مدیران خوب باشد این می‌تواند مفید باشد. هر چه قدر ما سر انتخاب اولیه‌امان شک و شبهه داشته باشیم، اگر رقابت بگذاریم در نزدیک یک ناحیه‌ای که نزدیک هم باشند performance نسبی به دست می‌آید و فروشگاه‌ها نمی‌توانند یک‌ه تازی کنند. مثلاً به آن‌ها گفته می‌شود که ببین این فروشگاه نزدیک شما چه قدر بهتر عمل می‌کند. گاهی اوقات این رقابت که زیاد شود efficiency را بالا می‌برد. در بعضی از موارد این برعکس است. اختلاف بین دو دوست بعضی وقت‌ها باعث می‌شود که دو طرف پیشرفت کنند ولی ممکن است که باعث دلخوری شود. اختلاف ممکن است که بعضاً efficiency را کم کند یا زیاد کند. در فرهنگ غرب این قانون رقابت است که تو اگر بهتر کار کنی حتی اون پانصد متر بغلی هم مال تو ولی در شرق این گونه نیست.

گاهی اوقات کنار هم بودن فروشگاه‌ها می‌تواند هم‌افزایی هم ایجاد کند. آیا خوب است که دو لبنیاتی کنار هم باشد؟ لبنیاتی کنار هم باشد معمولاً مشکل ایجاد می‌شود ولی کتابفروشی‌ها کنار هم هستند و این هم‌افزایی ایجاد می‌کند.



استراتژی product چه بود؟ freshness. که این یک سیستم توزیع قوی می‌طلبد و گاهی حتی یک فروشگاه سه بار visit داشته است. مثلاً در تهران یک روز دو روز یک بار تحویل داده می‌شود. این خیلی عملیات سنگین است که روزی سه بار به یک فروشگاه سر زده شود. حتی در چیدمان عملیاتمان، به شدت اثر دارد. لجستیک به شدت قوی دارد و حتی در زمان زلزله از دولت و پلیس سریع‌تر عمل کرده بوده است.

به هر حال، اگر کسی می‌خواهد low price عمل کند. یکی بخواهد تنوع داشته باشد و variety مطرح باشد، باید یادمان باشد که یک economies of scale & scope لازم است تا به cost برسیم. مثلاً freshness مهم است ولی تا به حد ممکن است. هر کاری که بخواهیم بکنیم هزینه دارد و تا یک جایی مفید است و از آن جا به بعد دارم هزینه می‌دهم و آن gain لازم را ندارد.

ایران خودرو انبارها متعدد داشت ولی سایپا محدودیت داشت و چون محدودیت داشت، به طور efficeint تر عمل می‌کرد. وقتی محدودیت وجود داشته باشد، رشد ایجاد می‌شود و چابکی وجود دارد. یک آدم وقتی در کاله کار می‌کند، چون کالا خراب شدنی است و سه روزه باید از آمل به تهران برسد و در نتیجه چابک است. اگر این فرد وارد ایران خودرو شود، قطعاً دچار مشکل می‌شود.

یادگیری‌های کلیدی از این کیس: سیستم آی تی و زیرساخت آن بسیار مهم است. در یک کلان شهر، وجود یک سیستم آی تی با زیر ساخت قوی، بسیار مفید است و کسی که این‌ها را انجام دهد واقعاً به دردش می‌خورد. در ecommerce استاندارد بودن محصول بسیار حیاتی است. مثلاً نان یک محصول استاندارد نیست و این باعث مشکل می‌شود.

وقتی تعامل با بازار باشد، می‌توان گفت که مثلاً این بازار این نوع آدامس را دوست دارد و می‌تواند به تولید کننده بگوید که چه چیزی تولید کند. عملاً این کیس، یک distribution همراه با retailing داشت.

جنبه ی دیگر کیس، این است که نگاه global view را head quarter دارد و visibility را دارد. نگاه centralized و decentralized به خوبی وجود دارد. وقتی یک فروشگاه می‌گوید من یک چیز را می‌خواهم، hq و سیستم centralized چیزهایی را می‌بیند که این محصول به دردش می‌خورد و می‌تواند بفروشد یا نمی‌تواند بفروشد. وقتی فروشگاه صد عدد می‌خواهد، می‌دانم که به خاطر رعد و برق امروز، صد عدد فروش نمی‌رود. سوال این است که آیا باید این مقدار محصول را ارائه دهیم یا نباید ارائه دهیم. Centralized as much as possible, decentralized as much as needed. باید در عین حال که centralized عمل می‌کنیم، اما به صورت decentralized هم قدرت مانور داشته باشیم.

آیا خوب است که چیدمان محصولات را فروشگاه جابه جا کنیم؟ نگاه سنتی در بازاریابی این بود که هر از گاهی عوض کنیم تا مردم درگیر گشتن شوند و در فروشگاه وقت بیش‌تری صرف کنند. امروزه نگاه این است که مردم گرفتاراند و باید مطابق نیاز مشتری به گونه‌ای بگذاریم که سریع مشتری به محصولات دسترسی داشته باشد.

آخرین block، demand fulfillment بود.

CTP (Capable to Promise): گاهی اوقات ممکن است که یک سفارش وارد سیستم شود و به cube مربوطه مراجعه کنم. موجودی به این معنا است که بالقوه می‌توانم سفارش بگیرم. الآن زمان گذشته و یک

مشتری دیگر می‌آید و یک سفارش می‌خواهد. سوال این است که آیا به او قول بدهم یا ندهم. زمان این جا سریع است و مثلاً ATP صد عدد بوده است ولی سفارش ۱۲۰ می‌آید. اینجا به این فکر می‌افتیم که آیا می‌توان به گونه‌ای اون مقدار ۲۰ تا را تأمین کرد.

Dynamic Re-allocation: این که کدام تولید را به چه کسی assign کنم، می‌توانم reallocate کنم. این جا دیده می‌شود که یک سری flow برای مصرف و یک مقداری برای تولید وجود دارد. وقتی تقاضا کنسل می‌شود یا تغییر می‌کند می‌توانم مجدداً re-allocate کنم. رابطه‌ی تولید و عرضه را مجدد باید update کنم. تقاضای برگشت خورده را بدهم به inventory یا تولید؟ در واقع این همان، مفهوم demand netting است.

چالش‌ها:

آیا من به هر سفارش جواب مثبت بدهم یا نه؟ یک سری چیزها برای یک سری افراد رزرو کنم.

در سیستم‌های ایرلاین و... این وارد revenue management می‌شود و این مفهومی است که مطرح است.

این run-time کمی دارد و باید بسیار سریع پاسخ‌گو بود.

از لحاظ IT، مقداری memory based و یک سری چیزها cpubased است.

این‌ها ویژگی‌های سیستم‌های APS می‌باشد و باید ۲۴/۷ در حال اجرا باشد و همیشه up and running دارد.

## پایان

مدیریت عملیات – دکتر شیخ زاده – زمستان ۹۴ و بهار ۹۵

محمد قربانیان

[www.narenjiman.ir](http://www.narenjiman.ir)